Revenue Cloud Developer Guide

Version 63.0, Spring ’25

Last updated: May 2, 2025

© Copyright 2000–2025 Salesforce, Inc. All rights reserved. Salesforce is a registered trademark of Salesforce, Inc., as are other names and marks. Other marks appearing herein may be trademarks of their respective owners.

CONTENTS

Chapter 1: Get Started with Revenue Cloud Developer Resources 1

Chapter 2: Product Catalog Management 4

Product Catalog Management 5
- Product Catalog Management Standard Objects 5
- Product Catalog Management Fields on Standard Objects 54
- Product Catalog Management Business APIs 62
- Product Catalog Management Metadata API Types 193
- Product Catalog Management Tooling API Objects 197

Product Discovery 204
- Product Discovery Business APIs 204
- Product Discovery Standard Invocable Actions 272
- Product Discovery Metadata API Types 300

Chapter 3: Salesforce Pricing 304

Salesforce Pricing Standard Objects 305
- AttributeAdjustmentCondition 306
- AttributeBasedAdjRule 310
- AttributeBasedAdjustment 312
- AttributeDefinition 317
- BundleBasedAdjustment 322
- CostBook 327
- ContractItemPrice 329
- CostBookEntry 333
- PriceAdjustmentSchedule 336
- PriceAdjustmentTier 339
- PriceBook2 344
- PriceBookEntry 347
- PriceBookEntryDerivedPrice 351
- PricingAdjBatchJob 355
- PricingAdjBatchJobLog 359
- PricingAPIExecution 362
- PricingProcedureResolution 365
- PricingProcessExecution 367
- ProductPriceHistoryLog 370
- ProductPriceRange 372
- ProductSellingModel 374
- ProductSellingModelDataTranslation 377
- ProductSellingModelOption 378

Salesforce Pricing Business APIs . . . 381
Resources . . . 383
Request Bodies . . . 410
Response Bodies . . . 434
Salesforce Pricing Standard Invocable Actions . . . 465
Run Salesforce Headless Pricing Action . . . 466
Run Salesforce Pricing Action . . . 472
Salesforce Pricing Metadata API Types . . . 475
Flow for Salesforce Pricing . . . 475
IndustriesPricingSettings . . . 476
PricingActionParameters . . . 478
PricingRecipe . . . 480
Salesforce Pricing Tooling API Objects . . . 485
PricingActionParameters . . . 486
PricingProcedureOutputMap . . . 489
PricingRecipe . . . 492
PricingRecipeTableMapping . . . 494
ProcedureOutputResolution . . . 496
ProcedurePlanCriterion . . . 499
ProcedurePlanDefinition . . . 501
ProcedurePlanDefinitionVersion . . . 504
ProcedurePlanOption . . . 506
ProcedurePlanSection . . . 510
ProcedurePlanVariable . . . 512

Chapter 4: Product Configurator . . . 514

Product Configurator Standard Objects . . . 515
ExpressionSetConstraintObj . . . 515
Product Configuration Flow . . . 517
Product Configuration Flow Assignment . . . 518
Product Configuration Rule . . . 520
Product Configurator Business APIs . . . 523
Resources . . . 525
Request Bodies . . . 539
Response Bodies . . . 553

Chapter 5: Transaction Management . . . 623

Quote and Order Capture . . . 624
Quote and Order Capture Standard Objects . . . 624
Quote and Order Capture Fields on Standard Objects . . . 659
Quote and Order Capture Tooling API Objects . . . 673
Quote and Order Capture Platform Event . . . 675
Quote and Order Capture Business APIs . . . 680
Quote and Order Capture Apex Reference . . . 761

Quote and Order Capture Standard Invocable Actions . . . 925
Quote and Order Capture Metadata API Types . . . 931
Asset Lifecycle . . . 932
Asset Lifecycle Standard Objects . . . 932
Asset Lifecycle Standard Invocable Actions . . . 992
Asset Lifecycle Metadata API Types . . . 1002
Asset Lifecycle Business APIs . . . 1003

Chapter 6: Usage Management . . . 1014

Usage Management Standard Objects . . . 1015
ProductUsageGrant . . . 1016
TransactionJournal . . . 1021
TransactionUsageEntitlement . . . 1024
UnitOfMeasure . . . 1032
UnitOfMeasureClass . . . 1035
UsageBillingPeriodItem . . . 1037
UsageEntitlementAccount . . . 1043
UsageEntitlementBucket . . . 1047
UsageEntitlementEntry . . . 1050
UsageGrantRenewalPolicy . . . 1055
UsageGrantRolloverPolicy . . . 1057
UsagePrdGrantBindingPolicy . . . 1059
UsageRatableSummary . . . 1061
UsageResource . . . 1067
UsageResourceBillingPolicy . . . 1071
UsageSummary . . . 1073
Usage Management Standard Invocable Actions . . . 1078
Invoke Summary Creation Action . . . 1078
Process Consumption Overages Action . . . 1079
Refresh Usage Entitlement Bucket Action . . . 1081
Usage Management Business APIs . . . 1082
Resources . . . 1083
Response Bodies . . . 1086
Usage Management Metadata API Types . . . 1093
Flow for Usage Management . . . 1093

Chapter 7: Rate Management . . . 1095

Rate Management Standard Objects . . . 1096
PriceBookRateCard . . . 1096
RateAdjustmentByAttribute . . . 1098
RateAdjustmentByTier . . . 1103
RateCard . . . 1108
RateCardEntry . . . 1110
RatingFrequencyPolicy . . . 1115

RatingRequest . . . 1117
RatingRequestBatchJob . . . 1120
Rate Management Metadata API Types . . . 1122
IndustriesRatingSettings . . . 1122
Flow for Rate Management . . . 1124
Rate Management Business APIs . . . 1124
Resources . . . 1124
Response Bodies . . . 1126
Rate Management Standard Invocable Actions . . . 1130
Invoke Rating Service Action . . . 1130

Chapter 8: Dynamic Revenue Orchestrator . . . 1134

Dynamic Revenue Orchestrator Standard Objects . . . 1135
AssetFulfillmentDecomp . . . 1137
FulfillmentAsset . . . 1139
FulfillmentAssetAttribute . . . 1141
FulfillmentAssetRelationship . . . 1143
FulfillmentFalloutRule . . . 1145
FulfillmentLineAttribute . . . 1148
FulfillmentLineRel . . . 1150
FulfillmentLineSourceRel . . . 1153
FulfillmentOrder . . . 1155
FulfillmentOrderLineItem . . . 1170
FulfillmentPlan . . . 1177
FulfillmentStep . . . 1180
FulfillmentStepDefinition . . . 1191
FulfillmentStepDefinitionGroup . . . 1198
FulfillmentStepDependency . . . 1200
FulfillmentStepDependencyDef . . . 1201
FulfillmentStepJeopardyRule . . . 1203
FulfillmentStepSource . . . 1206
FulfillmentTaskAssignmentRule . . . 1208
FulfillmentWorkspace . . . 1210
FulfillmentWorkspaceItem . . . 1212
Order . . . 1213
OrderAction . . . 1233
ProductDecompEnrichmentRule . . . 1235
ProductFulfillmentDecompRule . . . 1239
ProductFulfillmentScenario . . . 1242
SalesTransactionFulfillReq . . . 1244
ValTfrm . . . 1248
ValTfrmGrp . . . 1251
Dynamic Revenue Orchestrator Standard Invocable Actions . . . 1254
Submit Order Action . . . 1254

Submit Sales Transaction Action . . . 1268
Dynamic Revenue Orchestrator Metadata API Types . . . 1272
Flow for Dynamic Revenue Orchestrator . . . 1272
DynamicFulfillmentOrchestratorSettings . . . 1273
Callouts in Dynamic Revenue Orchestrator . . . 1275
Configure Callout Settings . . . 1276
Default Order Fulfillment Provider . . . 1276
Apex Type Provider . . . 1277
External Services Defined Provider . . . 1282

Chapter 9: Billing . . . 1283

Billing Standard Objects . . . 1284
AccountBillingAccount . . . 1287
AccountingPeriod . . . 1287
BillingAccount . . . 1289
BillingBatchScheduler . . . 1290
BillingBatchFilterCriteria . . . 1297
BillingMilestonePlan . . . 1301
BillingMilestonePlanItem . . . 1303
BillingPeriodItem . . . 1308
BillingPolicy . . . 1312
BillingSchedule . . . 1314
BillingScheduleGroup . . . 1324
BillingTreatment . . . 1333
BillingTreatmentItem . . . 1336
BsgRelationship . . . 1341
CreditMemo . . . 1344
CreditMemoAddressGroup . . . 1352
CreditMemoInvApplication . . . 1353
CreditMemoLine . . . 1357
CreditMemoLineInvoiceLine . . . 1364
CreditMemoLineTax . . . 1369
GeneralLedgerAccount . . . 1375
GeneralLedgerAcctAsgntRule . . . 1377
InvBatchDraftToPostedRun . . . 1381
Invoice . . . 1384
InvoiceAddressGroup . . . 1394
InvoiceBatchRun . . . 1395
InvoiceBatchRunCriteria . . . 1403
InvoiceBatchRunRecovery . . . 1406
InvoiceDocument . . . 1408
InvoiceLine . . . 1410
InvoiceLineRelationship . . . 1420
InvoiceLineTax . . . 1424

LegalEntity . . . 1430
LegalEntyAccountingPeriod . . . 1432
PaymentTerm . . . 1435
PaymentTermItem . . . 1436
RevenueTransactionErrorLog . . . 1439
TaxEngine . . . 1442
TaxEngineProvider . . . 1446
TaxPolicy . . . 1447
TaxTreatment . . . 1449
TransactionJournal . . . 1452
Billing Platform Events . . . 1456
BillingScheduleCreatedEvent . . . 1456
CreditInvoiceProcessedEvent . . . 1460
CreditMemoProcessedEvent . . . 1464
InvoiceProcessedEvent . . . 1467
NegInvcLineProcessedEvent . . . 1472
VoidInvoiceProcessedEvent . . . 1475
Billing Standard Invocable Actions . . . 1478
Apply Credit Action . . . 1479
Create Billing Schedules From Billing Transaction Action . . . 1481
Generate Invoice Documents Action . . . 1483
Post Draft Invoice Action . . . 1484
Post Draft Invoice Batch Run Action . . . 1486
Recover Billing Schedules Action . . . 1488
Unapply Credit Action . . . 1489
Billing Business APIs . . . 1491
Billing Business API Limits . . . 1494
Resources . . . 1495
Request Bodies . . . 1539
Response Bodies . . . 1604
Billing Apex Reference . . . 1631
ConnectApi Namespace . . . 1632
TaxEngineAdapter Interface . . . 1659
Billing Metadata API Types . . . 1666
BillingSettings . . . 1666
Flow for Billing . . . 1669

Chapter 10: Revenue Cloud Associated Objects . . . 1670

StandardObjectNameChangeEvent . . . 1671
StandardObjectNameFeed . . . 1673
StandardObjectNameHistory . . . 1680
StandardObjectNameOwnerSharingRule . . . 1682
StandardObjectNameShare . . . 1684
Event . . . 1685

Task 1706

CHAPTER 1 Get Started with Revenue Cloud Developer Resources

Get a single, unified system to automate your CRM processes. Use the developer sources of Revenue Cloud to automate the backend work to support the end-to-end revenue solution.

Revenue Cloud provides extensible and API-first business components of the product-to-cash processes. Learn more about the developer resources that are available for these components.

EDITIONS

Available in: Lightning Experience

Available in: **Enterprise**, **Unlimited**, and **Developer** Editions

Product Catalog Management

Create and manage an entire product portfolio with components such as attributes, product classifications, simple and bundled products, and rules.

- Use standard objects and fields to manage products, rules, and catalogs.
- Use business APIs to serve catalog definitions to users or applications.
- Use metadata API types to access and manage the metadata types, such as product specification type and product specification record types.
- Use tooling API objects to retrieve and manage smaller pieces of metadata types through SOQL capabilities. Use REST or SOAP to access metadata.
- Use Product Discovery business APIs, which are composite APIs, to search products or to discover catalogs, products, and categories.

Salesforce Pricing

Create a reliable pricing solution for your users through customized price adjustment schedules. Get accurate pricing for your entire product portfolio.

- Use standard objects and fields to manage pricing processes such as product management, and the calculation and application of discounts.
- Use business APIs to get unified pricing experiences across product lines.
- Use invocable actions to invoke the pricing Connect API by providing the pricing, context, and price waterfall details.
- Use metadata API types to work with the metadata associated with Flows and Salesforce Pricing settings.

- Use tooling API objects to retrieve and manage smaller pieces of metadata types through SOQL capabilities such as pricing action parameters, pricing procedure output map, and pricing recipe details. Use REST or SOAP to access metadata.

Product Configurator

Customize the components and attributes of a product to meet the business requirement expectations.

- Use standard objects to manage product-related information.
- Use the business APIs to retrieve and update a product's configuration from a configurator or to access configurator capabilities by integrating with any front-end application.

Quote and Order Capture

Manage subscription lifecycles from quotes and orders to contracts, assets, amendments, and renewals.

- Use standard objects and fields to manage transactions. Use the QuoteSaveEvent platform event to notify subscribers after saving of a quote is processed.
- Use business APIs to place an order or a quote.
- Use invocable actions to create and activate an order from a quote.
- Use metadata API types to work with the metadata associated with Flows.
- Use built-in Apex classes and interfaces grouped by namespace.

Asset Lifecycle

Get insights into customer assets and see a consolidated list of all assets that belong to an account.

- Use standard objects to manage details of a customer asset.
- Use invocable actions to create or update an asset from an order or an order item. You can also initiate amendment, renewal, or cancellation of assets through invocable actions or business APIs.
- Use metadata API types to work with the metadata associated with Flows.

Usage Management

Ensure transparent, accurate, and efficient management of usage data and estimated usage amount.

- Use standard objects and fields to set up and manage consumption of usage-based products.
- Use metadata API types to work with the metadata associated with Usage Management.
- Use business APIs to get details of a usage-based product that's associated with an asset, an order item, or a quote line item.
- Use invocable actions to invoke usage summaries, process consumption overages, and refresh usage entitlements.

Rate Management

Quote and price products based on predefined rates for future use of the product or service.

- Use standard objects and fields to manage rates and discounts for a product's resource consumption.
- Use metadata API types to work with the metadata associated with Rate Management settings.
- Use business APIs to get details of a rate plan and persisted rating waterfall.
- Use invocable action to invoke the rating service to rate the usage records.

Dynamic Revenue Orchestrator

Get visibility into a product's fulfillment journey. Also, get a view of the entire fulfillment design processes.

- Use standard objects to manage details of a product's fulfillment.
- Use invocable actions to submit an order or a sales transaction to Dynamic Revenue Orchestrator for fulfillment.
- Use metadata API types to work with the metadata associated with Flows.
- Use callout step types to make HTTP calls to an external system.

Billing

Get an integrated and extensible subscription and usage-based billing solution. Automate processes such as payment processing, invoice generation, and usage-based billing.

- Use standard objects to manage billing and tax configurations, credit memos, and invoices.
- Use platform events types to know more about standard platform events.
- Use invocable actions to manage credit application, billing schedules, and invoices.
- Use business APIs to manage credit application and to handle billing scenarios.
- Use built-in Apex classes to access the same capabilities that are available in the Billing Business APIs.
- Use metadata API types to work with the metadata associated with Billing settings and Flows.

SEE ALSO:

Business Rules Engine

Context Service

Salesforce Contracts

CHAPTER 2 Product Catalog Management

In this chapter ...

- Product Catalog Management
- Product Discovery

Create and manage a product catalog with components, such as attributes, product classifications, simple and bundled products, and rules.

Product Catalog Management

Manage an entire product portfolio with components such as attributes, product classifications, simple and bundled products, and rules.

EDITIONS

Available in: Lightning Experience

Available in: **Enterprise**, **Unlimited**, and **Developer** Editions of Revenue Cloud

Product Catalog Management Standard Objects

The Product Catalog Management data model provides objects and fields to manage products, rules, and catalogs.

Product Catalog Management Fields on Standard Objects

Product Catalog Management adds standard and custom fields to some standard Salesforce objects. These fields are available only in orgs where Product Catalog Management is enabled. This object is available in API version 60.0 and later.

Product Catalog Management Business APIs

Use primitive APIs of Product Catalog Management that serve catalog definitions to users or applications.

Product Catalog Management Metadata API Types

Metadata API enables you to access some types and feature settings that you can customize in the user interface.

Product Catalog Management Tooling API Objects

Tooling API exposes metadata used in developer tooling that you can access through REST or SOAP. Tooling API's SOQL capabilities for many metadata types allow you to retrieve smaller pieces of metadata.

SEE ALSO:

Salesforce Help: Product Catalog Management Permission Set Licenses

Salesforce Help: User Permissions for Product Catalog Management

Product Catalog Management Standard Objects

The Product Catalog Management data model provides objects and fields to manage products, rules, and catalogs.

AttributeCategory

Represents a logical grouping of attributes that can be reused while defining products. Attribute Categories are used for searching and managing product attributes. For example, the "Mobile Handset Properties" category has color, storage and make model, and size attributes. This object is available in API version 60.0 and later.

AttributeCategoryAttribute

Represents a relationship between an attribute category and the attribute definition. This object is available in API version 60.0 and later.

AttrPicklistExcludedValue

Represents the excluded picklist values for a product classification attribute or a product attribute definition. This object is available in API version 61.0 and later.

ProductAttributeDefinition

Represents the relationship between a product and its attributes. This object is available in API version 60.0 and later.

ProductCategoryDisqual

Represents disqualification rules for product categories. The rules determine when the product category doesn't qualify to be displayed to users. This object is available in API version 60.0 and later.

ProductCategoryQualification

Represents qualification rules for product categories. The rules determine when the product category qualifies to be displayed to users. This object is available in API version 60.0 and later.

ProductClassification

Represents a template that holds a collection of dynamic attributes. Product classification is used to quickly define and create multiple products that are similar yet different. This object is available in API version 60.0 and later.

ProductClassificationAttr

Represents the relationship between a product classification and its attributes. This is the default configuration for products based on the product classification. This object is available in API version 60.0 and later.

ProductComponentGrpOverride

Represents override information for a Product Component Group. The cardinality of the product component group can be overridden in the context of a product bundle. This object is available in API version 60.0 and later.

ProductDisqualification

Represents disqualification rules for products. The rules determine when the product doesn't qualify to be displayed to users. The rules are based on user context. This object is available in API version 60.0 and later.

ProductQualification

Represents qualification rules for products. The rules determine when the product qualifies to be displayed to users. The rules are based on user context. This object is available in API version 60.0 and later.

ProductRampSegment

Represents the ramp period within a ramp deal where terms, volumes, and other commitments change over time. This object is available in API version 62.0 and later.

ProductRelComponentOverride

Represents the cardinality overrides for product components in a bundle. This object is available in API version 60.0 and later.

ProductSpecificationRecType

Represents the relationship between industry-specific product specifications and the product record type. This object is available in API version 60.0 and later.

ProductSpecificationType

Represents the type of product specification provided by the user to make the product terminology unique to an industry. A product specification type is associated with a product specification record type. This object is available in API version 60.0 and later.

SEE ALSO:

Object Reference for the Salesforce Platform: Overview of Salesforce Objects and Fields

SOAP API Developer Guide: Introduction to SOAP API

AttributeCategory

Represents a logical grouping of attributes that can be reused while defining products. Attribute Categories are used for searching and managing product attributes. For example, the "Mobile Handset Properties" category has color, storage and make model, and size attributes. This object is available in API version 60.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Product Catalog Management must be enabled to access this object.

Fields

Field	Details
`Code`	**Type** string **Properties** Create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The unique code for the attribute category. The maximum size is 80 alphanumeric characters. The code can include the following special characters: @ ! - < > * ? + = % # ( ) / \ & ‘ £ € $ ”.
`Description`	**Type** textarea **Properties** Create, Nillable, Update **Description** The description of the attribute category that's used only during design time.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the attribute category was last referenced.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the attribute category was last viewed.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** The unique name of the attribute category. The maximum length is 80 characters (of any type).

Field	Details
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The owner of the attribute category. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

AttributeCategoryFeed on page 1673
Feed tracking is available for the object.

AttributeCategoryHistory on page 1680
History is available for tracked fields of the object.

AttributeCategoryShare on page 1684
Sharing is available for the object.

AttributeCategoryAttribute

Represents a relationship between an attribute category and the attribute definition. This object is available in API version 60.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Product Catalog Management must be enabled to access this object.

Fields

Field	Details
AttributeCategoryId	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The ID of the attribute category that the attribute is associated with. The ID is unique within the organization. This field is a relationship field. **Relationship Name** AttributeCategory **Relationship Type** Lookup **Refers To** AttributeCategory
AttributeDefinitionId	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The ID of the attribute definition associated with the attribute category. The ID is unique within the organization. This field is a relationship field. **Relationship Name** AttributeDefinition **Relationship Type** Lookup **Refers To** AttributeDefinition
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the attribute category attribute was last referenced.
LastViewedDate	**Type** dateTime

Field	Details
	Properties Filter, Nillable, Sort **Description** The date the attribute category attribute was last viewed.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** A unique name for the attribute. The maximum length is 80 characters (of any type).
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The owner of the attribute category attribute. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

AttributeCategoryAttributeFeed on page 1673
Feed tracking is available for the object.

AttributeCategoryAttributeHistory on page 1680
History is available for tracked fields of the object.

AttributeCategoryAttributeShare on page 1684
Sharing is available for the object.

AttrPicklistExcludedValue

Represents the excluded picklist values for a product classification attribute or a product attribute definition. This object is available in API version 61.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Product Catalog Management must be enabled to access this object.

Fields

Field	Details
`AttributeId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The ID of the product classification attribute or the product attribute definition of the picklist data type. This field is a polymorphic relationship field. **Relationship Name** Attribute **Relationship Type** Lookup **Refers To** ProductAttributeDefinition, ProductClassificationAttr
`AttributePicklistValueId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The ID of the attribute picklist value that's excluded in the product classification attribute or product attribute definition. This field is a relationship field. **Relationship Name** AttributePicklistValue **Relationship Type** Lookup **Refers To** AttributePicklistValue

Field	Details
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the excluded attribute picklist value was last referenced.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the excluded attribute picklist value was last viewed.
Name	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** The name of the excluded attribute picklist value.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The owner of the excluded attribute picklist value. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

AttrPicklistExcludedValueFeed **on page 1673**

Feed tracking is available for the object.

AttrPicklistExcludedValueHistory on page 1680
History is available for tracked fields of the object.

AttrPicklistExcludedValueShare on page 1684
Sharing is available for the object.

ProductAttributeDefinition

Represents the relationship between a product and its attributes. This object is available in API version 60.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Product Catalog Management must be enabled to access this object.

Fields

Field	Details
`AttributeCategoryId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort **Description** The ID of the attribute category assigned to the parent object. This field is a relationship field. **Relationship Name** AttributeCategory **Relationship Type** Lookup **Refers To** AttributeCategory
`AttributeDefinitionId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The ID of the attribute associated with the product. This field is a relationship field.

Field	Details
	Relationship Name AttributeDefinition **Relationship Type** Lookup **Refers To** AttributeDefinition
`AttributeNameOverride`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The name to display for the attribute when shown for this object. This display name overrides the name on the attribute. For example, the attribute "Color" is overridden to display as "Laptop Color." The maximum size is 255 alphanumeric characters. The name can include these special characters: @ ! - < > * ? + = % # ( ) / \ & ' £ € $ ".
`DefaultValue`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The default value for the product attribute. The attribute value can be changed. This default overrides the default value set for a picklist for the attribute.
`Description`	**Type** textarea **Properties** Create, Nillable, Update **Description** A description of the attribute definition. The maximum size is 32,000 alphanumeric characters. The description can include these special characters: @ ! - < > * ? + = % # ( ) / \ & ' £ € $ ".
`DisplayType`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The type to display data for the selected data type. Possible values are: • `CheckBox`—Checkbox

Field	Details
	• `ComboBox`—Combobox • `Date` • `Datetime`—Date Time • `Number` • `RadioButton`—Radio Button • `Slider`—Available in API version 61.0 and later • `Text` • `Toggle`
`HelpText`	**Type** textarea **Properties** Create, Nillable, Update **Description** The help text to display when end users are configuring this attribute. This field overrides the help text defined for the attribute itself.
`IsHidden`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates if this product attribute is hidden from end users in the run time (`true`) or not (`false`). The default value is `false`.
`IsPriceImpacting`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether this attribute dictates the price of a product (`true`) or not (`false`). The default value is `false`.
`IsReadOnly`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the attribute is read-only for users in the run time (`true`) or not (`false`). The default value is `false`.

Field	Details
`IsRequired`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether this attribute requires a value when assigned to a parent object (`true`) or not (`false`). The default value is `false`.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the product attribute was last referenced.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the product attribute was last viewed.
`MaximumCharacterCount`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The maximum number of characters that can be entered for an attribute value.
`MaximumValue`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The maximum value that can be entered as an attribute value.
`MinimumCharacterCount`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The minimum number of characters that can be entered for an attribute value.
`MinimumValue`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The minimum value that can be entered as an attribute value.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** The name of the product attribute.
`OverriddenProductAttributeDefinitionId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort **Description** The ID associated with the overridden product attribute definition. This field is a relationship field. **Relationship Name** OverriddenProductAttributeDefinition **Relationship Type** Lookup **Refers To** ProductAttributeDefinition
`OverrideContextId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort **Description** The ID associated with the root product record in a bundle. This field is a polymorphic relationship field. **Relationship Name** OverrideContext

Field	Details
	Relationship Type Lookup **Refers To** Product2
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The owner of the product attribute definition. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User
`Product2Id`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The product ID associated with the product attribute definition. This field is a relationship field. **Relationship Name** Product2 **Relationship Type** Lookup **Refers To** Product2
`ProductClassificationAttributeId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The ID of the attribute assigned to the product classification. This field is a relationship field.

Field	Details
	Relationship Name ProductClassificationAttribute **Relationship Type** Lookup **Refers To** ProductClassificationAttr
`Sequence`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The display sequence of the attribute when configuring the product during run time.
`Status`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The lifecycle state of the product attribute definition. Valid values are: • `Active` • `Draft` • `Inactive` The default value is `Draft`.
`StepValue`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The increment or decrement by which a slider's value changes as the user adjusts the product attribute value. Available in API version 61.0 and later.
`ValueDescription`	**Type** textarea **Properties** Create, Nillable, Update **Description** The description of the value assigned to this attribute. This field takes on the value description from the attribute definition.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ProductAttributeDefinitionFeed **on page 1673**
Feed tracking is available for the object.

ProductAttributeDefinitionHistory **on page 1680**
History is available for tracked fields of the object.

ProductAttributeDefinitionShare **on page 1684**
Sharing is available for the object.

ProductCategoryDisqual

Represents disqualification rules for product categories. The rules determine when the product category doesn't qualify to be displayed to users. This object is available in API version 60.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Product Catalog Management must be enabled to access this object.

Fields

Field	Details
`CategoryId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The product category associated with the product disqualification record. This field is a relationship field. **Relationship Name** Category **Relationship Type** Lookup **Refers To** ProductCategory
`EffectiveFromDate`	**Type** date

Field	Details
	Properties Create, Filter, Group, Sort, Update **Description** The date from which the disqualification rule for the product category comes into effect.
`EffectiveToDate`	**Type** date **Properties** Create, Filter, Group, Sort, Update **Description** The date to which the disqualification rule for the product category ceases to be in effect.
`IsDisqualified`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Indicates whether the product category is disqualified (`true`) or not (`false`) based on the disqualification rules. The default value is `false`.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the product category disqualification record was last referenced.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the product category disqualification record was last viewed.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name of the product category disqualification record.

Field	Details
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The ID of the owner of the product category disqualification record. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User
Reason	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The reason to disqualify the product category.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ProductCategoryDisqualFeed on page 1673
Feed tracking is available for the object.

ProductCategoryDisqualHistory on page 1680
History is available for tracked fields of the object.

ProductCategoryDisqualShare on page 1684
Sharing is available for the object.

ProductCategoryQualification

Represents qualification rules for product categories. The rules determine when the product category qualifies to be displayed to users. This object is available in API version 60.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Product Catalog Management must be enabled to access this object.

Fields

Field	Details
`CategoryId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The product category associated with the category qualification record. This field is a relationship field. **Relationship Name** Category **Relationship Type** Lookup **Refers To** ProductCategory
`EffectiveFromDate`	**Type** date **Properties** Create, Filter, Group, Sort, Update **Description** The date from which the qualification rule for the product category comes into effect.
`EffectiveToDate`	**Type** date **Properties** Create, Filter, Group, Sort, Update **Description** The date to which the qualification rule for the product category ceases to be in effect.
`IsQualified`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Indicates whether the product category is qualified (`true`) or not (`false`) based on the qualification rules. The default value is `false`.

Field	Details
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the product category qualification record was last referenced.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the product category qualification record was last viewed.
Name	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name of the product category qualification record.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The ID of the owner of the product category qualification record. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ProductCategoryQualificationFeed on page 1673

Feed tracking is available for the object.

ProductCategoryQualificationHistory on page 1680
History is available for tracked fields of the object.

ProductCategoryQualificationShare on page 1684
Sharing is available for the object.

ProductClassification

Represents a template that holds a collection of dynamic attributes. Product classification is used to quickly define and create multiple products that are similar yet different. This object is available in API version 60.0 and later.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), undelete(), update(), upsert()`

Special Access Rules

Product Catalog Management must be enabled to access this object.

Fields

Field	Details
`Code`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** A unique code for the product classification. The maximum size is 80 alphanumeric characters. The code can include the following special characters: @ ! - < > * ? + = % # ( ) / \ & ' £ € $ ".
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the product classification record was last referenced.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the product classification record was last viewed.

Field	Details
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** The name of the product classification. The maximum length is 80 characters (of any type).
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The owner of the product classification. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User
`Status`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The lifecycle status of the product classification. Possible values are: • `Active` • `Draft` • `Inactive` The default value is `Draft`.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ProductClassificationFeed **on page 1673**
Feed tracking is available for the object.

ProductClassificationHistory on page 1680

History is available for tracked fields of the object.

Sharing rules are available for the object.

ProductClassificationShare on page 1684

Sharing is available for the object.

ProductClassificationAttr

Represents the relationship between a product classification and its attributes. This is the default configuration for products based on the product classification. This object is available in API version 60.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Product Catalog Management must be enabled to access this object.

Fields

Field	Details
`AttributeCategoryId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort **Description** The ID of the attribute category assigned to the parent object. This field is a relationship field. **Relationship Name** AttributeCategory **Relationship Type** Lookup **Refers To** AttributeCategory
`AttributeDefinitionId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The ID of the attribute assigned to the parent object.

Field	Details
	This field is a relationship field. **Relationship Name** AttributeDefinition **Relationship Type** Lookup **Refers To** AttributeDefinition
`AttributeNameOverride`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The overridden attribute name to display for the attribute when shown for this object. For example, "Color" overridden to "Laptop Color."
`DefaultValue`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The default value of the attribute for a product based on the product classification. This value can be changed.
`Description`	**Type** textarea **Properties** Create, Nillable, Update **Description** The description of this product classification attribute definition.
`DisplayType`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The type to display data for the selected data type. Valid values are: • `CheckBox`—Checkbox • `ComboBox`—Combobox • `Date`

Field	Details
	• `Datetime`—Date Time • `Number` • `RadioButton`—Radio Button • `Slider`—Available in API version 61.0 and later • `Text` • `Toggle`
`ExcludedPicklistValues`	**Type** textarea **Properties** Create, Nillable, Update **Description** The picklist values excluded from the attribute picklist. This field ensures that the product classification attribute only has valid values.
`HelpText`	**Type** textarea **Properties** Create, Nillable, Update **Description** The help text to display when end users are configuring this attribute. This field overrides the help text defined for the attribute itself.
`IsHidden`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates if this product attribute is hidden from end users in the run time (`true`) or not (`false`). The default value is `false`.
`IsPriceImpacting`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether this attribute dictates the price of a product (`true`) or not (`false`). The default value is `false`.

Field	Details
`IsReadOnly`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the attribute is read-only for users in the run time (`true`) or not (`false`). The default value is `false`.
`IsRequired`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether this attribute requires a value when assigned to a parent object (`true`) or not (`false`). The default value is `false`.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the product classification attribute was last referenced.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the product classification attribute was last viewed.
`MaximumCharacterCount`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The maximum number of characters that can be entered for an attribute value.
`MaximumValue`	**Type** string

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update **Description** The maximum value that can be entered as an attribute value.
`MinimumCharacterCount`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The minimum number of characters that can be entered for an attribute value.
`MinimumValue`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The minimum value that can be entered as an attribute value.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** The name of the product classification attribute.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The ID of the product classification attribute owner. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User

Field	Details
`ProductClassificationId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The ID of the product classification that the attribute is associated with. This field is unique within your organization. This field is a relationship field. **Relationship Name** ProductClassification **Relationship Type** Lookup **Refers To** ProductClassification
`Sequence`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The display sequence of the attribute when configuring the product during run time.
`Status`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The lifecycle status of the product classification attribute. Valid values are: • `Active` • `Draft` • `Inactive` The default value is `Draft`.
`StepValue`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The increment or decrement by which a slider's value changes as the user adjusts the product classification attribute value. Available in API version 61.0 and later.
`ValueDescription`	**Type** textarea **Properties** Create, Nillable, Update **Description** The description of the value assigned to this attribute. This field takes on the value description from the attribute definition.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ProductClassificationAttrFeed on page 1673
Feed tracking is available for the object.

ProductClassificationAttrHistory on page 1680
History is available for tracked fields of the object.

Sharing rules are available for the object.

ProductClassificationAttrShare on page 1684
Sharing is available for the object.

ProductComponentGrpOverride

Represents override information for a Product Component Group. The cardinality of the product component group can be overridden in the context of a product bundle. This object is available in API version 60.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Product Catalog Management must be enabled to access this object.

Fields

Field	Details
`IsExcluded`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the product component group is excluded from the product bundle in the runtime. Excluding a group automatically excludes all child components of the group. The default value is `false`.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the product component override record was last referenced.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the product component override record was last viewed.
`MaxBundleComponents`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The maximum number of components that can be added to a group.
`MinBundleComponents`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The minimum number of components that must be added to a group.
`Name`	**Type** string

Field	Details
	Properties Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name of the overridden product component group.
`OverrideContextId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The root bundle product in whose context the group cardinality is overridden. This field is a polymorphic relationship field. **Relationship Name** OverrideContext **Relationship Type** Lookup **Refers To** Product2
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The owner of the product component group override record. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User
`ProductComponentGroupId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The ID of the product component group record. This field is a relationship field.

Field	Details
	Relationship Name ProductComponentGroup **Relationship Type** Lookup **Refers To** ProductComponentGroup

ProductDisqualification

Represents disqualification rules for products. The rules determine when the product doesn't qualify to be displayed to users. The rules are based on user context. This object is available in API version 60.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Product Catalog Management must be enabled to access this object.

Fields

Field	Details
`EffectiveFromDate`	**Type** date **Properties** Create, Filter, Group, Sort, Update **Description** The date from which the disqualification rule for the product comes into effect.
`EffectiveToDate`	**Type** date **Properties** Create, Filter, Group, Sort, Update **Description** The date to which the disqualification rule for the product ceases to be in effect.
`IsDisqualified`	**Type** boolean

Field	Details
	Properties Defaulted on create, Filter, Group, Sort **Description** Indicates whether the product is disqualified based on the disqualification rules (`true`) or not (`false`). The default value is `false`.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the product disqualification record was last referenced.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the product disqualification record was last viewed.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name of the product disqualification record.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The owner of the product disqualification record. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User

Field	Details
ParentProductId	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ID of the immediate parent product in the product bundle hierarchy. This field is a relationship field. **Relationship Name** ParentProduct **Relationship Type** Lookup **Refers To** Product2
ProductId	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The product for which the disqualification rule is defined. This field is a relationship field. **Relationship Name** Product **Relationship Type** Lookup **Refers To** Product2
Reason	**Type** textarea **Properties** Create, Nillable, Update **Description** The reason to disqualify the product.
RootProductId	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ID of the root product in the product bundle hierarchy.

Field	Details
	This field is a relationship field. **Relationship Name** RootProduct **Relationship Type** Lookup **Refers To** Product2

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ProductDisqualificationFeed on page 1673
Feed tracking is available for the object.

ProductDisqualificationHistory on page 1680
History is available for tracked fields of the object.

ProductDisqualificationShare on page 1684
Sharing is available for the object.

ProductQualification

Represents qualification rules for products. The rules determine when the product qualifies to be displayed to users. The rules are based on user context. This object is available in API version 60.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Product Catalog Management must be enabled to access this object.

Fields

Field	Details
`EffectiveFromDate`	**Type** date **Properties** Create, Filter, Group, Sort, Update **Description** The date from which the qualification rule for the product comes into effect.

Field	Details
`EffectiveToDate`	**Type** date **Properties** Create, Filter, Group, Sort, Update **Description** The date to which the qualification rule for the product ceases to be in effect.
`IsQualified`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Indicates whether the product is qualified based on the qualification rules (`true`) or not (`false`). For a product to qualify, this field should be true. The default value is `false`.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the product qualification record was last referenced.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the product qualification record was last viewed.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name of the product qualification record.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update

Field	Details
	Description The owner of the product qualification record. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User
`ParentProductId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ID of the immediate parent product in the product bundle hierarchy. This field is a relationship field. **Relationship Name** ParentProduct **Relationship Type** Lookup **Refers To** Product2
`ProductId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The product for which the qualification rule is defined. This field is a relationship field. **Relationship Name** Product **Relationship Type** Lookup **Refers To** Product2
`RootProductId`	**Type** reference

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update **Description** The ID of the root product in the product bundle hierarchy. This field is a relationship field. **Relationship Name** RootProduct **Relationship Type** Lookup **Refers To** Product2

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ProductQualificationFeed **on page 1673**
Feed tracking is available for the object.

ProductQualificationHistory **on page 1680**
History is available for tracked fields of the object.

ProductQualificationShare **on page 1684**
Sharing is available for the object.

ProductRampSegment

Represents the ramp period within a ramp deal where terms, volumes, and other commitments change over time. This object is available in API version 62.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Product Catalog Management must be enabled to access this object.

Fields

Field	Details
`DurationType`	**Type** picklist

Field	Details
	Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The unit of time within which users can try the product for free. Valid values are: • `Days` • `Months`
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the product ramp segment was last referenced.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the product ramp segment was last viewed.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name of the product ramp segment.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The ID of the owner of the product ramp segment. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User

Field	Details
`ProductId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The ID of the product associated with the product ramp segment. This field is a relationship field. **Relationship Name** Product **Refers To** Product2
`ProductSellingModelId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The ID of the product selling model associated with the product ramp segment. This field is a relationship field. **Relationship Name** ProductSellingModel **Refers To** ProductSellingModel
`SegmentType`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update **Description** The time duration within the ramp deal where specific terms, volumes, and commitments are applied to the subscription product. Valid values are: • `Custom` • `FreeTrial` • `Yearly` The default value is `Yearly`.
`TrialDuration`	**Type** int

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update **Description** The duration within which users can try the product for free.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ProductRampSegmentFeed on page 1673
Feed tracking is available for the object.

ProductRampSegmentHistory on page 1680
History is available for tracked fields of the object.

ProductRampSegmentShare on page 1684
Sharing is available for the object.

ProductRelComponentOverride

Represents the cardinality overrides for product components in a bundle. This object is available in API version 60.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Product Catalog Management must be enabled to access this object.

Fields

Field	Details
`DoesBundlePriceIncludeChild`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the bundle price includes the associated child component's price (`true`) or not (`false`). The default value is `false`.

Field	Details
`IsComponentRequired`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the component is a required component in the product bundle. The default value is `false`.
`IsDefaultComponent`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether this component is included in the product component group by default. The default value is `false`.
`IsExcluded`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the bundle excludes the component (`true`) or not (`false`). The default value is `false`.
`IsQuantityEditable`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the product component quantity can be edited (`true`) or not (`false`). The default value is `false`.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the product related component override record was last referenced.
`LastViewedDate`	**Type** dateTime

Field	Details
	Properties Filter, Nillable, Sort **Description** The date the product related component override record was last viewed.
`MaxQuantity`	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The maximum quantity for the product component in the product bundle.
`MinQuantity`	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The minimum quantity for the product component in the product bundle
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name of the product related component override. The maximum length is 255 characters (of any type).
`OverrideContextId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The ID associated with the root product in a bundle. This field is a polymorphic relationship field. **Relationship Name** OverrideContext **Relationship Type** Lookup **Refers To** Product2

Field	Details
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The owner ID of the product related component override record. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User
ProductRelatedComponentId	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The ID associated with the product related component record. This field is a relationship field. **Relationship Name** ProductRelatedComponent **Relationship Type** Lookup **Refers To** ProductRelatedComponent
Quantity	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The default number of child product related components.
QuantityScaleMethod	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field	Details
	Description The scaling method that determines how the child product quantity changes as the quantity of the parent product changes in the runtime cart. Possible values are: • `Constant` • `Proportional` The default value is `Proportional`.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ProductRelComponentOverrideFeed **on page 1673**
Feed tracking is available for the object.

ProductRelComponentOverrideHistory **on page 1680**
History is available for tracked fields of the object.

ProductRelComponentOverrideShare **on page 1684**
Sharing is available for the object.

ProductSpecificationRecType

Represents the relationship between industry-specific product specifications and the product record type. This object is available in API version 60.0 and later.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

Product Catalog Management must be enabled to access this object.

Fields

Field	Details
`DeveloperName`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** The unique name of the ProductSpecificationRecType object in the API. The name:

Field	Details
	• must be 40 characters or fewer. • can contain only underscores and alphanumeric characters. • must begin with a letter. • can contain only underscores and alphanumeric characters. • can't include spaces • can't end with an underscore • can't contain 2 consecutive underscores
`IsCommercial`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the product is sold commercially (`true`) or not (`false`)
`Language`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The combined language and locale ISO code, which controls the language of the Product Specification Record Type. Possible values are: • `da`—Danish • `de`—German • `en_US`—English • `es`—Spanish • `es_MX`—Spanish (Mexico) • `fi`—Finnish • `fr`—French • `it`—Italian • `ja`—Japanese • `ko`—Korean • `nl_NL`—Dutch • `no`—Norwegian • `pt_BR`—Portuguese (Brazil) • `ru`—Russian • `sv`—Swedish • `th`—Thai

Field	Details
	• `zh_CN`—Chinese (Simplified) • `zh_TW`—Chinese (Traditional)
`MasterLabel`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** Label for this Product Specification Record Type value. This display value is the internal label that doesn't get translated.
`NamespacePrefix`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The namespace prefix that is associated with this object. Each Developer Edition org that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the ***`namespacePrefix_componentName`*** The namespace prefix can have one of the following values. • In Developer Edition orgs, `NamespacePrefix` is set to the namespace prefix of the org for all objects that support it, unless an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition org of the package developer. • In orgs that aren't Developer Edition orgs, `NamespacePrefix` is set only for objects that are part of an installed managed package. All other objects have no namespace prefix.
`ProductSpecificationType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The product specification type that's associated with the record type.
`RecordTypeId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update

Field	Details
	Description The ID of the record type that's associated with the product specification type. This field is a relationship field. **Relationship Name** RecordType **Relationship Type** Lookup **Refers To** RecordType

ProductSpecificationType

Represents the type of product specification provided by the user to make the product terminology unique to an industry. A product specification type is associated with a product specification record type. This object is available in API version 60.0 and later.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

Product Catalog Management must be enabled to access this object.

Fields

Field	Details
`Description`	**Type** textarea **Properties** Create, Nillable, Update **Description** The description of the Product Specification Type.
`DeveloperName`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** The unique name of the Product Specification Type object in the API. The name: • must be 40 characters or fewer. • can contain only underscores and alphanumeric characters.

Field	Details
	• must begin with a letter. • can contain only underscores and alphanumeric characters. • can't include spaces • can't end with an underscore • can't contain 2 consecutive underscores In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization.
`Language`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The combined language and locale ISO code, which controls the language of the Product Specification Type. Possible values are: • `da`—Danish • `de`—German • `en_US`—English • `es`—Spanish • `es_MX`—Spanish (Mexico) • `fi`—Finnish • `fr`—French • `it`—Italian • `ja`—Japanese • `ko`—Korean • `nl_NL`—Dutch • `no`—Norwegian • `pt_BR`—Portuguese (Brazil) • `ru`—Russian • `sv`—Swedish • `th`—Thai • `zh_CN`—Chinese (Simplified) • `zh_TW`—Chinese (Traditional)
`MasterLabel`	**Type** string **Properties** Create, Filter, Group, Sort, Update

Field	Details
	Description Label for this Product Specification Type value. This display value is the internal label that doesn't get translated.
`NamespacePrefix`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The namespace prefix that is associated with this object. Each Developer Edition org that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the ***`namespacePrefix_componentName`*** The namespace prefix can have one of the following values. • In Developer Edition orgs, `NamespacePrefix` is set to the namespace prefix of the org for all objects that support it, unless an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition org of the package developer. • In orgs that aren't Developer Edition orgs, `NamespacePrefix` is set only for objects that are part of an installed managed package. All other objects have no namespace prefix.

Product Catalog Management Fields on Standard Objects

Product Catalog Management adds standard and custom fields to some standard Salesforce objects. These fields are available only in orgs where Product Catalog Management is enabled. This object is available in API version 60.0 and later.

Product Catalog Management Fields on Attribute Definition

Standard and custom fields extend the standard Attribute Definition object for use in Product Catalog Management.

Product Catalog Management Fields on Product2

Standard and custom fields extend the standard Product2 object for use in Product Catalog Management to represent information about products.

Product Catalog Management Fields on Product Catalog

Standard and custom fields extend the standard Product Catalog object for use in Product Catalog Management. This object is available in API version 60.0 and later.

Product Catalog Management Fields on Product Category

Standard and custom fields extend the standard Product Category object for use in Product Catalog Management.

Product Catalog Management Fields on Product Component Group

Standard and custom fields extend the standard Product Component Group object for use in Product Catalog Management.

Product Catalog Management Fields on Product Related Component

Standard and custom fields extend the standard Product Related Component object for use in Product Catalog Management.

Product Catalog Management Fields on Product Relationship Type

Standard and custom fields extend the standard Product Relationship Type object for use in Product Catalog Management.

Product Catalog Management Fields on Product Selling Model Option

Standard and custom fields extend the standard Product Selling Model Option object for use in Product Catalog Management. This object is available in API version 60.0 and later.

Product Catalog Management Fields on Attribute Definition

Standard and custom fields extend the standard Attribute Definition object for use in Product Catalog Management.

Fields

Field	Details
`DataType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort **Description** The data type of the attribute definition. Valid values are: • `Checkbox` • `Currency`—Available in API version 61.0 and later • `Date` • `Datetime` • `Number` • `Percent`—Available in API version 61.0 and later • `Picklist` • `Text`

SEE ALSO:

Attribute Definition

Product Catalog Management Fields on Product2

Standard and custom fields extend the standard Product2 object for use in Product Catalog Management to represent information about products.

Fields

Field	Details
`Based On`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ID of the product classification from which this product inherits. This field is a relationship field. **Relationship Name** BasedOn **Relationship Type** Lookup **Refers To** ProductClassification
`Help Text`	**Type** textarea **Properties** Create, Nillable, Update **Description** The help text that appears at runtime for the product. The maximum size is 32,000 alphanumeric characters. The help text can include these special characters: @ ! - < > * ? + = % # ( ) / \ & ' £ € $ ".
`Availability Date`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date when the product is available.
`CanRamp`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Indicates whether the product's terms, volumes, and other commitments can be ramped (true) at run time or not (false) The default value is `false`.

Field	Details
`Discontinued Date`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date when the product is discontinued.
`End Of Life Date`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date and time after which a product isn't supported, ordered, or maintained.
`Specification Type`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The type of product specification that's being created.
`DecompositionScope`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The number of fulfillment order line items that must be generated. Available in API version 61.0 and later. Valid values are: • `Account` • `Bundle` • `Order` • `OrderLineItem`
`FulfillmentQtyCalcMethod`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Determines whether the quantity of fulfillment order line items must always be one or must be aggregated from the source line items. Available in API version 61.0 and later.

Field	Details
	Valid values are: • `Aggregate` • `AlwaysOne`
`UsageModelType`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The type of usage model for a product or service. Anchor is the main subscription product or service. Pack is the add-on product or service that grants additional usage resources for consumption. Available in API version 62.0 and later. Valid values are: • `Anchor` • `Pack`

SEE ALSO:

Product2

Product Catalog Management Fields on Product Catalog

Standard and custom fields extend the standard Product Catalog object for use in Product Catalog Management. This object is available in API version 60.0 and later.

Fields

Field	Details
`Code`	**Type** string **Properties** Create, Filter, Group, idLookup, Nillable, Sort, Update **Description** A unique ID associated with the catalog. The maximum size is 80 alphanumeric characters.
`Description`	**Type** textarea **Properties** Create, Nillable, Update

Field	Details
	Description The description of the catalog that's used during design time. The maximum size is 255 alphanumeric characters.
`EffectiveEndDate`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date after which the product catalog is unavailable to end users.
`EffectiveStartDate`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date on which the product catalog is available to end users.
`CatalogType`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The category of an entry in the catalog. Possible values are: • `Sales` • `ServiceProcess`—Service Process The default value is `Sales`.

SEE ALSO:

Product Catalog

Product Catalog Management Fields on Product Category

Standard and custom fields extend the standard Product Category object for use in Product Catalog Management.

Fields

Field	Details
`Code`	**Type** string **Properties** Create, Filter, Group, idLookup, Nillable, Sort, Update **Description** A unique ID associated with the catalog. The maximum size is 80 alphanumeric characters.
`IsNavigational`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the category or subcategory is shown in the menu as a navigational breadcrumb (`true`) or not (`false`). Available in API version 62.0 and later. The default value is `false`.

SEE ALSO:

Product Category

Product Catalog Management Fields on Product Component Group

Standard and custom fields extend the standard Product Component Group object for use in Product Catalog Management.

Fields

Field	Details
`ParentGroupId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The parent product component group in a nested group hierarchy for the same parent product. Available in API version 62.0 and later. This field is a relationship field. **Relationship Name** ParentGroup

Field	Details
	Refers To ProductComponentGroup

SEE ALSO:

Product Component Group

Product Catalog Management Fields on Product Related Component

Standard and custom fields extend the standard Product Related Component object for use in Product Catalog Management.

Fields

Field	Details
`ChildProductClassificationId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort **Description** The child product classification that's associated with a product. This field is a relationship field. **Relationship Name** ChildProductClassification **Refers To** ProductClassification

SEE ALSO:

ProductRelatedComponent

Product Catalog Management Fields on Product Relationship Type

Standard and custom fields extend the standard Product Relationship Type object for use in Product Catalog Management.

Fields

Field	Details
`AssociatedProductRoleCat`	**Type** picklist

Field	Details
	Properties Create, Filter, Group, Nillable, Restricted picklist, Sort **Description** The role that the associated component plays in the relationship. Valid values are: • `BundleComponent`— The associated product is part of a bundle. • `ClassificationComponent`— The associated component is a product classification. Available in API version 61.0 and later

SEE ALSO:

Product Relationship Type

Product Catalog Management Fields on Product Selling Model Option

Standard and custom fields extend the standard Product Selling Model Option object for use in Product Catalog Management. This object is available in API version 60.0 and later.

Fields

Field	Details
`IsDefault`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates the default product selling model for a product. Setting a product selling model as default is optional. A product can only have one default product selling model. The default value is `false`.

SEE ALSO:

Product Selling Model Option

Product Catalog Management Business APIs

Use primitive APIs of Product Catalog Management that serve catalog definitions to users or applications.

This table lists the available Product Catalog Management resources.

Resource	Description
`/connect/pcm/catalogs` (POST)	Retrieve, search, filter, or sort catalog records.
`/connect/pcm/catalogs/`***`catalogId`*** (GET)	Retrieve details of catalog records based on a catalog ID.
`/connect/pcm/catalogs/`***`catalogId`***`/categories` (GET)	Retrieve the root-level categories of a catalog based on a catalog ID, or subcategories based on a parent category. You can also search, filter, or sort the categories.
`/connect/pcm/categories/`***`categoryId`*** (GET)	Retrieve details of individual category records based on a category ID.
`/connect/pcm/products` (POST)	Retrieve products. You can also search, filter, or sort the products.
`/connect/pcm/products/`***`productId`*** (GET)	Retrieve details of individual product records or a bundle based on a product ID.
`/connect/pcm/products/bulk` (POST)	Retrieve details for multiple products.
`/connect/pcm/index/configurations` (GET, PUT)	Retrieve the saved index configurations. Additionally, you can persist the index configuration.
`/connect/pcm/relatedRecords/`***`entityName`*** (POST)	Retrieve related ProductRampSegment or ProductUsageGrant records for Product2 object.
`/connect/pcm/index/snapshots` (GET)	Retrieve the created snapshots and snapshot indexes.
`/connect/pcm/index/deploy` (POST)	Create indexes for a snapshot. Indexes improve search results and make it easier to find products at run time through search terms.
`/connect/pcm/index/setting` (GET, PATCH)	Fetch and update settings related to indexing and search.

Resource	Description
`/connect/pcm/index/error` (GET)	Get the count and details of the errors that occurred during the indexing process.
`/connect/pcm/deep-clone` (POST)	Copy related records of an object along with the main product record.
`/connect/pcm/unit-of-measure/info` (GET)	Get details about the unit of measure for a specific set of records.
`/connect/pcm/unit-of-measure/rounded-data` (POST)	Round off and scale decimal data for a specific set of fields.

Resources
Learn more about the available Product Catalog Management API resources.

Request Bodies
Learn more about the available Product Catalog Management API request bodies.

Response Bodies
Learn more about the available Product Catalog Management API response bodies.

SEE ALSO:

Connect REST API Developer Guide: Introduction

Resources

Learn more about the available Product Catalog Management API resources.

Bulk Product Details (POST)
Retrieve details for multiple products.

Catalog List (POST)
Retrieve, search, filter, or sort catalog records.

Catalog By ID (GET)
Retrieve details of catalog records based on a catalog ID.

Categories List (GET)
Retrieve the root-level categories of a catalog based on a catalog ID, or subcategories based on a parent category. You can also search, filter, or sort the categories.

Category By ID (GET)
Retrieve details of individual category records based on a category ID.

Deep Clone (POST)
Copy related records of an object along with the main product record.

Index Configuration Collection (GET, PUT)
Retrieve the saved index configurations. Additionally, you can persist the index configuration.

Index Setting (GET, PATCH)
Fetch and update settings related to indexing and search.

Product Details (GET)
Retrieve details of individual product records or a bundle based on a product ID.

Product Related Records List (POST)
Retrieve related ProductRampSegment or ProductUsageGrant records for Product2 object.

Products List (POST)
Retrieve products. You can also search, filter, or sort the products.

Snapshot Collection (GET)
Retrieve the created snapshots and snapshot indexes.

Snapshot Deployment (POST)
Create indexes for a snapshot. Indexes improve search results and make it easier to find products at run time through search terms.

Snapshot Index Error (GET)
Get the count and details of the errors that occurred during the indexing process.

Unit of Measure Info (GET)
Get details about the unit of measure for a specific set of records.

Unit of Measure Rounded Data (POST)
Round off and scale decimal data for a specific set of fields.

Bulk Product Details (POST)

Retrieve details for multiple products.

Resource

```
/connect/pcm/products/bulk
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/pcm/products/bulk
```

Available version

61.0

HTTP methods

POST

Request body for POST

JSON example

```
{
  "correlationId": "cbfabffb-093f-45b3-8c2d-88b4acbd4867",
  "productIds": [
    "01tT1000000F0afIAC",
    "01tT1000000F0afIAC"
  ],
  "uptoLevel": 1,
```

```
  "additionalFields": {
    "Product2": {
      "fields": [
        "code__c"
      ]
    },
    "ProductAttributeDefinition": {
      "fields": [
        "scope"
      ]
    }
  }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`additionalFields`	Map<String, Additional Fields Input>	Map of object and list of additional standard or custom fields to be included in the response. The supported objects are: • Product2 • ProductAttributeDefinition—If the fields defined for the ProductAttributeDefinition object aren't available for the ProductClassificationAttr object, then the API request fails.	Optional	61.0
`correlationId`	String	Unique token to track and associate related events or transactions across different components of the application. If unspecified, a Universally Unique Identifier (UUID) is generated.	Optional	61.0
`productIds`	String[]	List of product IDs that details must be returned for. If any product ID is blank, invalid, or not found, then the request is processed with valid and available product IDs.	Required	61.0
`uptoLevel`	Integer	Hierarchy level to follow to return the product details. For a bundle, this property determines the number of levels of child components to be returned. You can specify up to a hierarchy level of 1.	Optional	61.0

Name	Type	Description	Required or Optional	Available Version
		If unspecified, the default level is the full bundle hierarchy.		

Response body for POST

Products Output

Catalog List (POST)

Retrieve, search, filter, or sort catalog records.

Resource

```
/connect/pcm/catalogs
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/pcm/catalogs
```

Available version

60.0

Requires Chatter

No

HTTP methods

POST

Note: The POST method is used to retrieve the catalog records instead of the GET method as a request payload is sent to filter the records.

Request body for POST

JSON example

This example shows how to retrieve catalogs that contain `apple` in the catalog name.

```
{
"pageSize": 100,
"offset": 0,
"filter": {
"criteria": [
{
"property": "name",
"operator": "contains",
"value": "apple"
}
]
}
}
```

This example shows how to retrieve catalogs with `ServiceProcess` as the catalog type.

```
{
"pageSize": 100,
```

```
"offset": 0,
"sort": {
  "orders": [
   {
   "property": "name",
   "direction": "desc"
   }
  ]
 },
"filter": {
"criteria": [
{
"property": "catalogType",
"operator": "eq",
"value": "ServiceProcess"
}
]
}
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`correlationId`	String	Unique token to track and associate related events or transactions across different components of the application. If unspecified, a Universally Unique Identifier (UUID) is generated.	Optional	60.0
`filter`	Filter	Criteria to filter the records. Filters are applicable to the fields of the ProductCatalog object. The supported operators are: • `eq` • `in` • `contains` The supported properties are `name` and `catalogType`.	Optional	60.0
`offset`	Integer	Number of records to skip. The default value is 0.	Optional	60.0
`pageSize`	Integer	Number of records per page. Valid values are from 1 through 100. If unspecified, defaults to 100.	Optional	60.0
`sort`	Sort	Sort order of the catalog records. The supported operators are: • `asc`	Optional	60.0

Name	Type	Description	Required or Optional	Available Version
		• desc		

Response body for POST

Catalogs Output

Catalog By ID (GET)

Retrieve details of catalog records based on a catalog ID.

Resource

```
/connect/pcm/catalogs/catalogId
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/pcm/catalogs/0ZST100000000kUOAQ
```

Available version

60.0

Requires Chatter

No

HTTP methods

GET

Request parameters for GET

Parameter Name	Type	Description	Required or Optional	Available Version
correlationId	String	Unique token to track and associate related events or transactions across different components of the application. If unspecified, a Universally Unique Identifier (UUID) is generated.	Optional	60.0
fields	String[]	For internal use only.	Optional	60.0

Response body for GET

Catalogs Output

Categories List (GET)

Retrieve the root-level categories of a catalog based on a catalog ID, or subcategories based on a parent category. You can also search, filter, or sort the categories.

Resource

```
/connect/pcm/catalogs/catalogId/categories
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/pcm/catalogs/0ZST100000000kUOAQ/categories
```

Available version

60.0

Requires Chatter

No

HTTP methods

GET

Request parameters for GET

Parameter Name	Type	Description	Required or Optional	Available Version
`correlationId`	String	Unique token to track and associate related events or transactions across different components of the application. If unspecified, a Universally Unique Identifier (UUID) is generated.	Optional	60.0
`depth`	Integer	Number of levels in the category hierarchy to return. The default value is 1.	Optional	60.0
`fields`	String[]	For internal use only.	Optional	60.0
`parentCategoryId`	String	ID of the category to fetch the associated hierarchy of subcategories. If unspecified, then the root-level categories are returned.	Optional	60.0

Response body for GET

Categories Output

Category By ID (GET)

Retrieve details of individual category records based on a category ID.

Resource

```
/connect/pcm/categories/categoryId
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/pcm/categories/0ZGT100000000qqOAA
```

Available version

60.0

Requires Chatter

No

HTTP methods

GET

Request parameters for GET

Parameter Name	Type	Description	Required or Optional	Available Version
`correlationId`	String	Unique token to track and associate related events or transactions across different components of the application. If unspecified, a Universally Unique Identifier (UUID) is generated.	Optional	60.0
`fields`	String[]	For internal use only.	Optional	60.0

Response body for GET

Categories Output

Deep Clone (POST)

Copy related records of an object along with the main product record.

Resource

```
/connect/pcm/deep-clone
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/pcm/deep-clone
```

Available version

63.0

HTTP methods

POST

Request body for POST

JSON example

```
{
  "mainRecordId": "01tSG0000028kcSYAQ",
  "mainObjectApiName": "Product2",
  "mainRecordFieldValues": {
    "Name": "New Cloud Storage"
  }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`mainObjectApiName`	String	API name of the object. The supported object is Product2.	Required	63.0

Name	Type	Description	Required or Optional	Available Version
`mainRecordFieldValues`	Map<String, String>	Mapping of the API name of the field to its value. The values passed through this map are set for the created record. You can pass the Name field only through this map.	Optional	63.0
`mainRecordId`	String	ID of the record.	Required	63.0

Response body for POST

Deep Clone Response

Index Configuration Collection (GET, PUT)

Retrieve the saved index configurations. Additionally, you can persist the index configuration.

Resource

```
/connect/pcm/index/configurations
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/pcm/index/configurations
```

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/pcm/index/configurations?includeMetadata=false&fieldTypes=Standard,Custom
```

Available version

62.0

HTTP methods

GET, PUT

Request parameters for GET

Parameter Name	Type	Description	Required or Optional	Available Version
`correlationId`	String	Unique token to track and associate related events or transactions across different components of the application. If unspecified, a Universally Unique Identifier (UUID) is generated.	Optional	62.0
`fieldTypes`	String[]	Filters and returns only the persisted index configurations, based on the index configuration type specified in the query parameters. The supported types of filters are: • `STANDARD` • `CUSTOM`	Optional	62.0

Parameter Name	Type	Description	Required or Optional	Available Version
		• `ProductDynamicAttribute` • `ProductAttributeDefinitionStandard` • `ProductAttributeDefinitionCustom`		
`includeMetadata`	Boolean	Indicates whether to include metadata (`true`) or not (`false`).	Optional	62.0

Response body for GET

Index Configuration Collection

Request body for PUT

JSON example

```
{
  "correlationId": "8545b5aa-f3e6-429a-8f21-9cc4ce50b1d7",
  "indexConfigurations": [
    {
      "attributeDefinitionId": "0tjT1000000002bIAA",
      "name": "Color",
      "type": "ProductDynamicAttribute",
      "isSearchable": true
    },
    {
      "attributeFieldId": "00Nxx000001FwnABII",
      "name": "Message__c",
      "type": "Custom",
      "isSearchable": true
    },
    {
      "name": "Code",
      "type": "Standard",
      "isSearchable": true
    },
    {
      "facetDisplayRank": 1,
      "isFacetable": false,
      "isSearchable": true,
      "name": "Family",
      "type": "Standard"
    }
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`correlationId`	String	Unique token to track and associate related events or transactions across different components of the application. If unspecified, a Universally Unique Identifier (UUID) is generated.	Optional	62.0
`indexConfigurations`	Index Configuration Input[]	List of index configurations.	Required	62.0

Response body for PUT

Index Configurations Update

Index Setting (GET, PATCH)

Fetch and update settings related to indexing and search.

Resource

```
/connect/pcm/index/setting
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/pcm/index/setting
```

Available version

63.0

HTTP methods

GET, PATCH

Response body for GET

Index Setting Results

Request body for PATCH

JSON example

```
{
    "setting" : {
        "supportedLanguages" : ["en_US","ja","es","nl_NL"],
        "defaultLanguage" : "en_US",
        "productsGrouping": "GROUPING_VARIATION"
   }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`setting`	Setting Input[]	Object containing the setting-related details.	Required	63.0

Request parameters for PATCH

Parameter Name	Type	Description	Required or Optional	Available Version
`settingId`	String	ID of the setting to update the details for.	Required	63.0

Response body for PATCH

Index Setting Update

Product Details (GET)

Retrieve details of individual product records or a bundle based on a product ID.

Resource

```
/connect/pcm/products/productId
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/pcm/products/01tT1000000F0afIAC
```

Available version

60.0

Requires Chatter

No

HTTP methods

GET

Note: You must invoke this API request by using GET method only. If the request is invoked by using POST method, the request is considered as a Products List API request.

Request parameters for GET

Parameter Name	Type	Description	Required or Optional	Available Version
`correlationId`	String	Unique token to track and associate related events or transactions across different components of the application. If unspecified, a Universally Unique Identifier (UUID) is generated.	Optional	60.0

Parameter Name	Type	Description	Required or Optional	Available Version
`fields`	String[]	For internal use only.	Optional	60.0

Response body for GET

Products

Product Related Records List (POST)

Retrieve related ProductRampSegment or ProductUsageGrant records for Product2 object.

Resource

```
/connect/pcm/relatedRecords/entityName
```

The supported entity or object is Product2.

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/pcm/relatedRecords/product2
```

Available version

62.0

HTTP methods

POST

Note: POST methods typically create an item, but for this resource POST is used to retrieve information.

Request body for POST

JSON example

```
{
  "recordIds": [
    "01txx0000006i44AAA",
    "01txx0000006i5gAAA"
  ],
  "relatedObjectNodes": [
    {
      "relatedObjectAPIName": "ProductRampSegment",
      "pageSize": 20,
      "offSet": 0
    },
    {
      "relatedObjectAPIName": "ProductUsageGrant",
      "pageSize": 10,
      "offSet": 0,
      "filter": {
        "criteria": [
          {
            "property": "status",
            "operator": "eq",
            "value": "active"
          },
```

```
          {
            "property": "effectivestartdate",
            "operator": "lte",
            "value": "2024-06-25"
          },
          {
            "criteriaType": "CustomWhereCondition",
            "value": "(effectiveenddate = null OR effectiveenddate >= 2024-06-25)"
          }
        ]
      }
    }
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`correlationId`	String	Unique token to track and associate related events or transactions across different components of the application. If unspecified, a Universally Unique Identifier (UUID) is generated.	Optional	62.0
`recordIds`	String[]	List of record IDs to return the relatedObjects records for. The maximum number of record IDs supported is 20.	Required	62.0
`relatedObjectNodes`	Related Object Node Input[]	List of nodes for the related objects. The maximum number of related object nodes supported is two.	Required	62.0

Response body for POST

Related Records List

Products List (POST)

Retrieve products. You can also search, filter, or sort the products.

Resource

```
/connect/pcm/products
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/pcm/products?productClassificationId=11BT10000004C9SMAU
```

Available version

60.0

Requires Chatter

No

HTTP methods

POST

Note: POST methods typically create an item, but for this resource POST is used to retrieve information.

Request body for POST

JSON example

This example is a search request for products that contain `Bundle Product` in the product name.

```
{
  "catalogIds": [
    "0ZST10000004D03OAE"
  ],
  "filter": {
    "criteria": [
      {
        "property": "name",
        "operator": "contains",
        "value": "Bundle Product"
      }
    ]
  },
  "relatedObjectFilters": [
    {
      "objectName": "ProductSpecificationRecType",
      "criteria": [
        {
          "property": "IsCommercial",
          "operator": "eq",
          "value": false
        }
      ],
      "additionalFields": {
        "Product2": {
          "fields": [
            "code__c"
          ]
        }
      }
    }
  ]
}
```

This example is a search request for products that contain the specified search term in the product name and are part of a catalog.

```
  {
  "catalogIds": [
    "0ZSDU0000002Og54AE"
  ],
  "searchTerm": "Slack"
}
```

Query parameters

Name	Type	Description	Required or Optional	Available Version
`productClassificationId`	String	ID of the product classification template that contains a collection of dynamic attributes and can be reused to create multiple products. Products that are based on a product classification inherit all the attributes of the product classification. Specify either the product classification ID, list of category IDs, or list of catalog IDs to retrieve products.	Optional	60.0

Properties

Name	Type	Description	Required or Optional	Available Version
`additionalFields`	Map<String, Additional Fields Input>	Map of the object and the list of standard and custom fields to be queried along with the standard response. The supported object is Product2.	Optional	61.0
`catalogIds`	List<String>	List of comma-separated catalog IDs. The API returns the list of product records that are associated with the specified catalog IDs. Specify either the product classification ID, list of category IDs, or list of catalog IDs to retrieve products.	Optional	60.0
`categoryIds`	List<String>	List of comma-separated category IDs. The API returns the list of product records that are associated with the specified category IDs. If unspecified, then returns all products that are added to at least one category. Specify either the product classification ID, list of category IDs, or list of catalog IDs to retrieve products.	Optional	60.0
`correlationId`	String	Unique token to track and associate related events or transactions across	Optional	60.0

Name	Type	Description	Required or Optional	Available Version
		different components of the application. If unspecified, a Universally Unique Identifier (UUID) is generated.		
`filter`	Criteria Input	Criteria to filter the records. Filters are applicable to the fields of Product2 object. The supported operators are: • `eq` • `in` • `contains`—This value isn't applicable if the **Use Indexed Data For Product Listing and Search** toggle from the Product Discovery Settings page from Setup is enabled. • `gt`—Specifies a greater than criteria. Available from API version 63.0 and later for Number, Date, and Datetime data types only. • `lt`—Specifies a less than criteria. Available from API version 63.0 and later for Number, Date, and Datetime data types only. • `gte`—Specifies a greater than or equal to criteria. Available from API version 63.0 and later for Number, Date, and Datetime data types only. • `lte`—Specifies a less than or equal to criteria. Available from API version 63.0 and later for Number, Date, and Datetime data types only. If multiple criteria are specified, then the criteria are combined by using the `and` operator. The supported properties are `name`, `description`, and `isActive`. If the **Use Indexed Data For Product Listing and Search** toggle from the Product Discovery Settings page from Setup is enabled, then the supported property is `name` only.	Optional	60.0
`offset`	Integer	Number of records to skip. The default value is 0.	Optional	60.0

Name	Type	Description	Required or Optional	Available Version
`pageSize`	Integer	Specifies the number of records per page. Valid values are from 1 through 200. If the value is unspecified, it defaults to 100.	Optional	60.0
`relatedObjectFilters`	Related Object Filter[]	Criteria for the related objects to filter the records. The supported operator is `eq`. The supported object is `ProductSpecificationRecType`. The supported values are `true` and `false`. The supported property is `IsCommercial`.	Optional	60.0
`searchTerm`	String	String used to get products with the product name containing the search term. See Search Considerations When Using Indexed Data.	Optional	62.0
`sort`	Sort	Sort order for the products. If the **Use Indexed Data For Product Listing and Search** toggle from the Product Discovery Settings page from Setup is enabled, then you can sort products by using name only.	Optional	60.0

Response body for POST

Products Output

Snapshot Collection (GET)

Retrieve the created snapshots and snapshot indexes.

Resource

```
/connect/pcm/index/snapshots
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/pcm/index/snapshots
```

Available version

62.0

HTTP methods

GET

Query Parameter for GET

Parameter Name	Type	Description	Required or Optional	Available Version
`numberOf IndexLogs`	Integer	Number of index logs to include in the response. Specify a number from 0 through 100. The default value is 25.	Optional	63.0

Response body for GET

Snapshot Collection

Snapshot Deployment (POST)

Create indexes for a snapshot. Indexes improve search results and make it easier to find products at run time through search terms.

Resource

```
/connect/pcm/index/deploy
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/pcm/index/deploy
```

Available version

62.0

HTTP methods

POST

Request body for POST

JSON example

This example shows a sample request to build a new snapshot with immediate activation.

```
{
  "snapshot": {
    "activationType": "IMMEDIATE"
  },
  "buildType": "FULL"
}
```

This example shows a sample request to rebuild a snapshot in the `active` status.

```
{
  "snapshot": {
    "activationType": "IMMEDIATE",
    "id": "1Avxx0000005DFe1AM"
  },
  "buildType": "FULL"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`buildType`	String	Build type of the snapshot index. Valid value is: • `FULL`—Specifies a full index build. • `INCREMENTAL`—Specifies an incremental index build. Available from API version 63.0 and later.	Required	62.0
`snapshot`	Run-time Catalog Snapshot Input[]	Snapshot to deploy.	Required	62.0

Response body for POST

Snapshot Deployment

Snapshot Index Error (GET)

Get the count and details of the errors that occurred during the indexing process.

Resource

```
/connect/pcm/index/error
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/pcm/index/error
```

Available version

63.0

HTTP methods

GET

Request parameters for GET

Parameter Name	Type	Description	Required or Optional	Available Version
`indexId`	String	ID of the index.	Required	63.0
`snapshotIndexId`	String	ID of the snapshot index.	Required	63.0

Response body for GET

Snapshot Index Error

Unit of Measure Info (GET)

Get details about the unit of measure for a specific set of records.

Resource

```
/connect/pcm/unit-of-measure/info
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/pcm/unit-of-measure/info
```

Available version

63.0

HTTP methods

GET

Query parameters for GET

Parameter Name	Type	Description	Required or Optional	Available Version
`correlationId`	String	Unique token to track and associate related events or transactions across different components of the application. If unspecified, a Universally Unique Identifier (UUID) is generated.	Optional	63.0
`ids`	String	IDs of the unit of measure records.	Optional	63.0

Response body for GET

Bulk Unit Of Measure Info

Unit of Measure Rounded Data (POST)

Round off and scale decimal data for a specific set of fields.

Resource

```
/connect/pcm/unit-of-measure/rounded-data
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/pcm/unit-of-measure/rounded-data
```

Available version

63.0

HTTP methods

POST

Request body for POST

JSON example

```
{
  "dataRowInputs": [
    {
      "key": "PRC1",
      "fieldDataInputs": [
```

```
        {
          "fieldApiName": "MaxQuantity",
          "originalValue": 1234.5678,
          "unitOfMeasureId": "0hExx0000000001EAA"
        },
        {
          "fieldApiName": "MinQuantity",
          "originalValue": "987462848934739347.32232590183756545",
          "unitOfMeasureId": "0hExx000000001dEAA"
        }
      ]
    },
    {
      "key": "PRC2",
      "fieldDataInputs": [
        {
          "fieldApiName": "MaxQuantity",
          "originalValue": 1234.5678,
          "unitOfMeasureId": "uomId1"
        },
        {
          "fieldApiName": "MinQuantity",
          "originalValue": 987.4628,
          "unitOfMeasureId": "Kgs Id"
        }
      ]
    },
    {
      "key": "PRC3",
      "fieldDataInputs": [
        {
          "fieldApiName": "MaxQuantity",
          "originalValue": 0.437584,
          "unitOfMeasureId": "uomId2"
        },
        {
          "fieldApiName": "MinQuantity",
          "originalValue": 7364.58923,
          "unitOfMeasureId": "uomId2"
        }
      ]
    }
  ]
}
```


Properties

Name	Type	Description	Required or Optional	Available Version
`correlationId`	String	Unique token to track and associate related events or transactions across different components of the application.	Optional	63.0

Name	Type	Description	Required or Optional	Available Version
		If unspecified, a Universally Unique Identifier (UUID) is generated.		
`dataRowInputs`	Data Row Input[]	List of row inputs for rounding the data.	Required	63.0

Response body for POST

Data Rounding

Request Bodies

Learn more about the available Product Catalog Management API request bodies.

Additional Fields Input

Input representation of the additional standard or custom fields to be included in the response.

Bulk Product Details Input

Input representation of the request to retrieve the details of multiple products.

Catalog Input

Input representation of the request to retrieve catalog records.

Criteria Input

Input representation of the filter criteria item request.

Data Rounding Input

Input representation of the details of the data rounding input.

Data Row Input

Input representation of the details of the input for a data rounding request.

Deep Clone Input

Input representation of the details of the object and associated record to be cloned.

Field Data Input

Input representation of the details of the field data input.

Filter Input

Input representation of the filter request.

Include Object Input

Input representation of the object to include in the response.

Index Configuration Collection Input

Input representation of the collection of index configurations.

Index Configuration Input

Input representation of the request to persist the index configuration.

Index Setting Input

Input representation of the index setting.

Options Input
Reserved for internal use.

Order Input
Input representation of the sort order item request.

Product Input
Input representation of a product in the catalog.

Related Object Filters Input
Input representation of the request to filter related objects.

Related Object Node Input
Input representation of the details of a related object node.

Related Records Input
Input representation of the request to retrieve related ProductRampSegment or ProductUsageGrant records for Product2 object.

Run-time Catalog Snapshot Input
Input representation of the details of a run-time catalog snapshot for deployment.

Setting Input
Input representation of the details of the index setting.

Snapshot Deployment Input
Input representation of the request to deploy a run-time catalog snapshot.

Sort Input
Input representation of the sort request.

Additional Fields Input

Input representation of the additional standard or custom fields to be included in the response.

JSON example

```
"additionalFields": {
  "Product2": {
    "fields": [
      "code__c"
    ]
  }
```

Properties

Name	Type	Description	Required or Optional	Available Version
`fields`	String[]	List of additional standard or custom fields to be included in the response.	Required	61.0

Bulk Product Details Input

Input representation of the request to retrieve the details of multiple products.

JSON example

```
{
  "correlationId": "cbfabffb-093f-45b3-8c2d-88b4acbd4867",
  "productIds": [
    "01tT1000000F0afIAC",
    "01tT1000000F0afIAC"
  ],
  "uptoLevel": 1,
  "additionalFields": {
    "Product2": {
      "fields": [
        "code__c"
      ]
    },
    "ProductAttributeDefinition": {
      "fields": [
        "scope"
      ]
    }
  }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
additionalFields	Map<String, Additional Fields Input>	Map of object and list of additional standard or custom fields to be included in the response. The supported objects are: • Product2 • ProductAttributeDefinition—If the fields defined for the ProductAttributeDefinition object aren't available for the ProductClassificationAttr object, then the API request fails.	Optional	61.0
correlationId	String	Unique token to track and associate related events or transactions across different components of the application. If unspecified, a Universally Unique Identifier (UUID) is generated.	Optional	61.0
productIds	String[]	List of product IDs that details must be returned for. If any product ID is blank, invalid, or not found, then the request is processed with valid and available product IDs.	Required	61.0

Name	Type	Description	Required or Optional	Available Version
`uptoLevel`	Integer	Hierarchy level to follow to return the product details. For a bundle, this property determines the number of levels of child components to be returned. You can specify up to a hierarchy level of 1. If unspecified, the default level is the full bundle hierarchy.	Optional	61.0

Catalog Input

Input representation of the request to retrieve catalog records.

JSON example

This example shows how to retrieve catalogs that contain `apple` in the catalog name.

```
{
"pageSize": 100,
"offset": 0,
"filter": {
"criteria": [
{
"property": "name",
"operator": "contains",
"value": "apple"
}
]
}
}
```

This example shows how to retrieve catalogs with `ServiceProcess` as the catalog type.

```
{
"pageSize": 100,
"offset": 0,
"sort": {
  "orders": [
   {
   "property": "name",
   "direction": "desc"
   }
  ]
 },
"filter": {
"criteria": [
{
"property": "catalogType",
"operator": "eq",
"value": "ServiceProcess"
}
]
```

```
}
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`correlationId`	String	Unique token to track and associate related events or transactions across different components of the application. If unspecified, a Universally Unique Identifier (UUID) is generated.	Optional	60.0
`filter`	Filter	Criteria to filter the records. Filters are applicable to the fields of the ProductCatalog object. The supported operators are: • `eq` • `in` • `contains` The supported properties are `name` and `catalogType`.	Optional	60.0
`offset`	Integer	Number of records to skip. The default value is 0.	Optional	60.0
`pageSize`	Integer	Number of records per page. Valid values are from 1 through 100. If unspecified, defaults to 100.	Optional	60.0
`sort`	Sort	Sort order of the catalog records. The supported operators are: • `asc` • `desc`	Optional	60.0

Criteria Input

Input representation of the filter criteria item request.

JSON example

```
"criteria":
[{
   "attributeType": "ProductStandard",
   "property": "name",
   "operator": "eq",
   "value": "iPhone"
},
```

```
{
   "criteriaType": "CustomWhereCondition",
   "value": "(effectiveenddate = null OR effectiveenddate >= 2024-06-25)"
}
]
```

Properties

Name	Type	Description	Required or Optional	Available Version
`attributeType`	String	Search attribute type of the facet for a faceted search. Valid values are: • `ProductStandard` • `ProductCustom` • `ProductDynamicAttribute` • `ProductAttributeStandard` • `ProductAttributeCustom` This property is applicable if the **Use Indexed Data For Product Listing and Search** toggle from the Product Discovery Settings page from Setup is enabled.	Optional	63.0
`criteriaType`	String	Type of criteria for the filter. Valid value is: • `CustomWhereCondition`	Required	60.0
`operator`	String	Operator used for the filter criteria. The supported operators are: • `eq` • `in` • `contains`—This value isn't applicable if the **Use Indexed Data For Product Listing and Search** toggle from the Product Discovery Settings page from Setup is enabled. • `gt`—Specifies a greater than criteria. Available from API version 63.0 and later for Number, Date, and Datetime data types only. • `lt`—Specifies a less than criteria. Available from API version 63.0 and later for Number, Date, and Datetime data types only. • `gte`—Specifies a greater than or equal to criteria. Available from API	Required	60.0

Name	Type	Description	Required or Optional	Available Version
		version 63.0 and later for Number, Date, and Datetime data types only. • `lte`—Specifies a less than or equal to criteria. Available from API version 63.0 and later for Number, Date, and Datetime data types only.		
`property`	String	Property name to use in the filter, which must be the same as the object field.	Required	60.0
`value`	Object	Value for the filter criteria.	Required	60.0

Data Rounding Input

Input representation of the details of the data rounding input.

JSON example

```
{
  "dataRowInputs": [
    {
      "key": "PRC1",
      "fieldDataInputs": [
        {
          "fieldApiName": "MaxQuantity",
          "originalValue": 1234.5678,
          "unitOfMeasureId": "0hExx0000000001EAA"
        },
        {
          "fieldApiName": "MinQuantity",
          "originalValue": "987462848934739347.32232590183756545",
          "unitOfMeasureId": "0hExx000000001dEAA"
        }
      ]
    },
    {
      "key": "PRC2",
      "fieldDataInputs": [
        {
          "fieldApiName": "MaxQuantity",
          "originalValue": 1234.5678,
          "unitOfMeasureId": "uomId1"
        },
        {
          "fieldApiName": "MinQuantity",
          "originalValue": 987.4628,
          "unitOfMeasureId": "Kgs Id"
        }
      ]
    },
```

```
    {
      "key": "PRC3",
      "fieldDataInputs": [
        {
          "fieldApiName": "MaxQuantity",
          "originalValue": 0.437584,
          "unitOfMeasureId": "uomId2"
        },
        {
          "fieldApiName": "MinQuantity",
          "originalValue": 7364.58923,
          "unitOfMeasureId": "uomId2"
        }
      ]
    }
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`correlationId`	String	Unique token to track and associate related events or transactions across different components of the application. If unspecified, a Universally Unique Identifier (UUID) is generated.	Optional	63.0
`dataRowInputs`	Data Row Input[]	List of row inputs for rounding the data.	Required	63.0

Data Row Input

Input representation of the details of the input for a data rounding request.

JSON example

```
JSON example
{
  "dataRowInputs": [
    {
      "key": "PRC1",
      "fieldDataInputs": [
        {
          "fieldApiName": "MaxQuantity",
          "originalValue": 1234.5678,
          "unitOfMeasureId": "0hExx0000000001EAA"
        },
        {
          "fieldApiName": "MinQuantity",
          "originalValue": "987462848934739347.32232590183756545",
          "unitOfMeasureId": "0hExx000000001dEAA"
        }
      ]
```

```
    },
    {
      "key": "PRC2",
      "fieldDataInputs": [
        {
          "fieldApiName": "MaxQuantity",
          "originalValue": 1234.5678,
          "unitOfMeasureId": "uomId1"
        },
        {
          "fieldApiName": "MinQuantity",
          "originalValue": 987.4628,
          "unitOfMeasureId": "Kgs Id"
        }
      ]
    },
    {
      "key": "PRC3",
      "fieldDataInputs": [
        {
          "fieldApiName": "MaxQuantity",
          "originalValue": 0.437584,
          "unitOfMeasureId": "uomId2"
        },
        {
          "fieldApiName": "MinQuantity",
          "originalValue": 7364.58923,
          "unitOfMeasureId": "uomId2"
        }
      ]
    }
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`fieldDataInputs`	Field Data Input[]	List of field-level data inputs.	Required	63.0
`key`	String	Key that identifies a unique data row.	Required	63.0

Deep Clone Input

Input representation of the details of the object and associated record to be cloned.

JSON example

```
{
  "mainRecordId": "01tSG0000028kcSYAQ",
  "mainObjectApiName": "Product2",
  "mainRecordFieldValues": {
```

```
    "Name": "New Cloud Storage"
  }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
mainObjectApiName	String	API name of the object. The supported object is Product2.	Required	63.0
mainRecordFieldValues	Map<String, String>	Mapping of the API name of the field to its value. The values passed through this map are set for the created record. You can pass the Name field only through this map.	Optional	63.0
mainRecordId	String	ID of the record.	Required	63.0

Field Data Input

Input representation of the details of the field data input.

JSON example

```
      "fieldDataInputs": [
        {
          "fieldApiName": "MaxQuantity",
          "originalValue": 0.437584,
          "unitOfMeasureId": "uomId2"
        },
        {
          "fieldApiName": "MinQuantity",
          "originalValue": 7364.58923,
          "unitOfMeasureId": "uomId2"
        }
      ]
```

Properties

Name	Type	Description	Required or Optional	Available Version
fieldApiName	String	Unique API name of the field.	Required	63.0
originalValue	String	Original value of the fields.	Required	63.0
unitOfMeasureId	String	ID of the unit of measure record that's associated to the field.	Required	63.0

Filter Input

Input representation of the filter request.

JSON example

```
"filter":
{
"criteria": [ {
"property": "name",
"operator": "eq",
"value": "iPhone"
},
{
"criteriaType": "CustomWhereCondition",
"value": "(effectiveenddate = null OR effectiveenddate >= 2024-06-25)"
}
]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`criteria`	Criteria[]	Details of the filter criteria.	Required if the `filter` property is specified.	60.0

Include Object Input

Input representation of the object to include in the response.

JSON example

```
"includeObjects":
[{
"objectName": "ProductCategory"
}]
```

Properties

Name	Type	Description	Required or Optional	Available Version
`objectName`	String	Name of the object to include in the response. The supported object is `ProductCategory`.	Required if the `options` property is specified.	60.0

Index Configuration Collection Input

Input representation of the collection of index configurations.

JSON example

```
{
  "correlationId": "8545b5aa-f3e6-429a-8f21-9cc4ce50b1d7",
```

```
  "indexConfigurations": [
    {
      "attributeDefinitionId": "0tjT1000000002bIAA",
      "name": "Color",
      "type": "ProductDynamicAttribute",
      "isSearchable": true
    },
    {
      "attributeFieldId": "00Nxx000001FwnABII",
      "name": "Message__c",
      "type": "Custom",
      "isSearchable": true
    },
    {
      "name": "Code",
      "type": "Standard",
      "isSearchable": true
    },
    {
      "facetDisplayRank": 1,
      "isFacetable": false,
      "isSearchable": true,
      "name": "Family",
      "type": "Standard"
    }
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`correlationId`	String	Unique token to track and associate related events or transactions across different components of the application. If unspecified, a Universally Unique Identifier (UUID) is generated.	Optional	62.0
`indexConfigurations`	Index Configuration Input[]	List of index configurations.	Required	62.0

Index Configuration Input

Input representation of the request to persist the index configuration.

JSON example

```
  "indexConfigurations": [
    {
      "attributeDefinitionId": "0tjT1000000002bIAA",
      "name": "Color",
      "type": "ProductDynamicAttribute",
      "isSearchable": true
```

```
    },
    {
      "attributeFieldId": "00Nxx000001FwnABII",
      "name": "Message__c",
      "type": "Custom",
      "isSearchable": true
    },
    {
      "name": "Code",
      "type": "Standard",
      "isSearchable": true
    },
    {
      "facetDisplayRank": 1,
      "isFacetable": false,
      "isSearchable": true,
      "name": "Family",
      "type": "Standard"
    }
  ]
```

Properties

Name	Type	Description	Required or Optional	Available Version
`attributeDefinitionId`	String	ID of the attribute definition.	Required if the `attributeFieldId` property isn't specified.	62.0
`attributeFieldId`	String	ID of the attribute field.	Required if the `attributeDefinitionId` property isn't specified.	62.0
`facetDisplayRank`	Integer	Sort order for displaying the facets at run time.	Optional	63.0
`isFacetable`	Boolean	Indicates whether the field is facetable (`true`) or not (`false`).	Optional	63.0
`isSearchable`	Boolean	Indicates whether the index-configured field is searchable (`true`) or not (`false`).	Optional	62.0
`name`	String	Name of the index-configured field.	Required	62.0
`type`	String	Type of the index-configured field.	Required	62.0

Index Setting Input

Input representation of the index setting.

JSON example

```
{
    "setting" : {
        "supportedLanguages" : ["en_US","ja","es","nl_NL"],
        "defaultLanguage" : "en_US",
        "productsGrouping": "GROUPING_VARIATION"
   }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`setting`	Setting Input[]	Object containing the setting-related details.	Required	63.0

Options Input

Reserved for internal use.

Order Input

Input representation of the sort order item request.

JSON example

```
"sort":{
"orders":
[{
   "property": "name",
   "direction": "asc"
}]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`direction`	String	Direction to sort the list items, such as in ascending order or descending order.	Required	60.0
`property`	String	Property to use for the sorting of the list items.	Required	60.0

Product Input

Input representation of a product in the catalog.

JSON example

This example is a search request for products that contain `Bundle Product` in the product name.

```
{
  "catalogIds": [
    "0ZST10000004D03OAE"
  ],
  "filter": {
    "criteria": [
      {
        "property": "name",
        "operator": "contains",
        "value": "Bundle Product"
      }
    ]
  },
  "relatedObjectFilters": [
    {
      "objectName": "ProductSpecificationRecType",
      "criteria": [
        {
          "property": "IsCommercial",
          "operator": "eq",
          "value": false
        }
      ],
      "additionalFields": {
        "Product2": {
          "fields": [
            "code__c"
          ]
        }
      }
    }
  ]
}
```

This example is a search request for products that contain the specified search term in the product name and are part of a catalog.

```
  {
  "catalogIds": [
    "0ZSDU0000002Og54AE"
  ],
  "searchTerm": "Slack"
}
```

Query parameters

Name	Type	Description	Required or Optional	Available Version
`productClassificationId`	String	ID of the product classification template that contains a collection of dynamic attributes and can be reused to create	Optional	60.0

Name	Type	Description	Required or Optional	Available Version
		multiple products. Products that are based on a product classification inherit all the attributes of the product classification. Specify either the product classification ID, list of category IDs, or list of catalog IDs to retrieve products.		

Properties

Name	Type	Description	Required or Optional	Available Version
`additionalFields`	Map<String, Additional Fields Input>	Map of the object and the list of standard and custom fields to be queried along with the standard response. The supported object is Product2.	Optional	61.0
`catalogIds`	List<String>	List of comma-separated catalog IDs. The API returns the list of product records that are associated with the specified catalog IDs. Specify either the product classification ID, list of category IDs, or list of catalog IDs to retrieve products.	Optional	60.0
`categoryIds`	List<String>	List of comma-separated category IDs. The API returns the list of product records that are associated with the specified category IDs. If unspecified, then returns all products that are added to at least one category. Specify either the product classification ID, list of category IDs, or list of catalog IDs to retrieve products.	Optional	60.0
`correlationId`	String	Unique token to track and associate related events or transactions across different components of the application. If unspecified, a Universally Unique Identifier (UUID) is generated.	Optional	60.0

Name	Type	Description	Required or Optional	Available Version
`filter`	Criteria Input	Criteria to filter the records. Filters are applicable to the fields of Product2 object. The supported operators are: • `eq` • `in` • `contains`—This value isn't applicable if the **Use Indexed Data For Product Listing and Search** toggle from the Product Discovery Settings page from Setup is enabled. • `gt`—Specifies a greater than criteria. Available from API version 63.0 and later for Number, Date, and Datetime data types only. • `lt`—Specifies a less than criteria. Available from API version 63.0 and later for Number, Date, and Datetime data types only. • `gte`—Specifies a greater than or equal to criteria. Available from API version 63.0 and later for Number, Date, and Datetime data types only. • `lte`—Specifies a less than or equal to criteria. Available from API version 63.0 and later for Number, Date, and Datetime data types only. If multiple criteria are specified, then the criteria are combined by using the `and` operator. The supported properties are `name`, `description`, and `isActive`. If the **Use Indexed Data For Product Listing and Search** toggle from the Product Discovery Settings page from Setup is enabled, then the supported property is `name` only.	Optional	60.0
`offset`	Integer	Number of records to skip. The default value is 0.	Optional	60.0
`pageSize`	Integer	Specifies the number of records per page. Valid values are from 1 through 200. If the value is unspecified, it defaults to 100.	Optional	60.0

Name	Type	Description	Required or Optional	Available Version
`relatedObjectFilters`	Related Object Filter[]	Criteria for the related objects to filter the records. The supported operator is `eq`. The supported object is `ProductSpecificationRecType`. The supported values are `true` and `false`. The supported property is `IsCommercial`.	Optional	60.0
`searchTerm`	String	String used to get products with the product name containing the search term. See Search Considerations When Using Indexed Data.	Optional	62.0
`sort`	Sort	Sort order for the products. If the **Use Indexed Data For Product Listing and Search** toggle from the Product Discovery Settings page from Setup is enabled, then you can sort products by using name only.	Optional	60.0

Related Object Filters Input

Input representation of the request to filter related objects.

JSON example

```
"relatedObjectFilters":
[{
   "criteria": [
    {
    "property": "IsCommercial",
    "operator": "eq",
    "value": true
    }
  ],
   "objectName": "ProductSpecificationRecType"
 }]
```

Properties

Name	Type	Description	Required or Optional	Available Version
`criteria`	Criteria[]	Criteria to filter the related objects.	Required if the `relatedObjectFilters` property is specified.	60.0

Name	Type	Description	Required or Optional	Available Version
`objectName`	String	API name of the object that's related to the main object.	Required if the `relatedObjectFilters` property is specified.	60.0

Related Object Node Input

Input representation of the details of a related object node.

JSON example

```
"relatedObjectNodes": [
   {
     "relatedObjectAPIName": "ProductRampSegment",
     "pageSize": 20,
     "offSet": 0
   },
   {
     "relatedObjectAPIName": "ProductUsageGrant",
     "pageSize": 10,
     "offSet": 0,
     "filter": {
       "criteria": [
         {
           "property": "status",
           "operator": "eq",
           "value": "active"
         },
         {
           "property": "effectivestartdate",
           "operator": "lte",
           "value": "2024-06-25"
         },
         {
           "criteriaType": "CustomWhereCondition",
           "value": "(effectiveenddate = null OR effectiveenddate >= 2024-06-25)"
         }
       ]
     }
   }
 ]
```

Properties

Name	Type	Description	Required or Optional	Available Version
`filter`	Criteria[]	Criteria to filter records. The supported properties are: • `StartDate` • `EndDate`	Optional	62.0

Name	Type	Description	Required or Optional	Available Version
		• `Status` The supported operators are: • `eq` • `gte` • `lte` The supported related object is ProductUsageGrant. If multiple criteria are specified, then the resultant criteria are combined by using the `and` operator.		
`offSet`	Integer	Number of records to skip. The default value is 0.	Optional	62.0
`pageSize`	Integer	Number of records per page. Valid values are from 1 through 100. If unspecified, the default value is 100.	Optional	62.0
`relatedObjectAPIName`	String	API name of the related object to return the records for. The supported related objects are ProductRampSegment and ProductUsageGrant.	Required	62.0

Related Records Input

Input representation of the request to retrieve related ProductRampSegment or ProductUsageGrant records for Product2 object.

JSON example

```
{
  "recordIds": [
    "01txx0000006i44AAA",
    "01txx0000006i5gAAA"
  ],
  "relatedObjectNodes": [
    {
      "relatedObjectAPIName": "ProductRampSegment",
      "pageSize": 20,
      "offSet": 0
    },
    {
      "relatedObjectAPIName": "ProductUsageGrant",
      "pageSize": 10,
      "offSet": 0,
      "filter": {
        "criteria": [
          {
            "property": "status",
```

```
            "operator": "eq",
            "value": "active"
          },
          {
            "property": "effectivestartdate",
            "operator": "lte",
            "value": "2024-06-25"
          },
          {
            "criteriaType": "CustomWhereCondition",
            "value": "(effectiveenddate = null OR effectiveenddate >= 2024-06-25)"
          }
        ]
      }
    }
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`correlationId`	String	Unique token to track and associate related events or transactions across different components of the application. If unspecified, a Universally Unique Identifier (UUID) is generated.	Optional	62.0
`recordIds`	String[]	List of record IDs to return the relatedObjects records for. The maximum number of record IDs supported is 20.	Required	62.0
`relatedObjectNodes`	Related Object Node Input[]	List of nodes for the related objects. The maximum number of related object nodes supported is two.	Required	62.0

Run-time Catalog Snapshot Input

Input representation of the details of a run-time catalog snapshot for deployment.

JSON example

```
  "snapshot": {
    "activationType": "IMMEDIATE",
    "activationDate": "2024-05-06T05:12:59.000Z",
    "id": "1Avxx0000005DFe1AM"
  }
```

Properties

Name	Type	Description	Required or Optional	Available Version
`activationDate`	String	Activation date of the snapshot.	Optional	62.0
`activationType`	String	Activation type of the snapshot. Valid value is: • `IMMEDIATE`—Snapshot is activated immediately after a successful build.	Required	62.0
`id`	String	ID of the snapshot.	Required	62.0

Setting Input

Input representation of the details of the index setting.

JSON example

```
    "setting" : {
        "supportedLanguages" : ["en_US","ja","es","nl_NL"],
        "defaultLanguage" : "en_US"
   }
```

Properties

Name	Type	Description	Required or Optional	Available Version
`default Language`	String	Default language for the API.	Required	63.0
`supported Languages`	String[]	List of supported language locales for indexing.	Required	63.0

Snapshot Deployment Input

Input representation of the request to deploy a run-time catalog snapshot.

JSON example

This example shows a sample request to build a new snapshot with immediate activation.

```
{
  "snapshot": {
    "activationType": "IMMEDIATE"
  },
  "buildType": "FULL"
}
```

This example shows a sample request to rebuild a snapshot in the `active` status.

```
{
  "snapshot": {
```

```
    "activationType": "IMMEDIATE",
    "id": "1Avxx0000005DFe1AM"
  },
  "buildType": "FULL"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`buildType`	String	Build type of the snapshot index. Valid value is: • `FULL`—Specifies a full index build. • `INCREMENTAL`—Specifies an incremental index build. Available from API version 63.0 and later.	Required	62.0
`snapshot`	Run-time Catalog Snapshot Input[]	Snapshot to deploy.	Required	62.0

Sort Input

Input representation of the sort request.

JSON example

```
"sort":
{
"orders":
[{
"property": "name",
"direction": "asc"
}]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`orders`	Order[]	Details of the sort order.	Required if the `sort` property is specified.	60.0

Response Bodies

Learn more about the available Product Catalog Management API response bodies.

Attribute Category

Output representation of the attribute category.

Attribute Definition
Output representation of the attribute definition.

Attribute Picklist
Output representation of the attribute picklist.

Attribute Picklist Value
Output representation of the attribute picklist value.

Bulk Unit Of Measure Info
Output representation of the details of the unit of measure records along with error details.

Data Rounding
Output representation of the data rounding response.

Data Row
Output representation of the details of a data row.

Catalog Output
Output representation of the catalog definition.

Catalogs Output
Output representation of the retrieved catalog result.

Categories Output
Output representation of the retrieved categories result.

Category Output
Output representation of the category definition.

Deep Clone Error
Output representation of the error details related to the deep clone request.

Deep Clone Record Response
Output representation of the details of the cloned related records.

Deep Clone Response
Output representation of the details of the cloned record.

Error Output
Output representation of the error details.

Facet Value
Output representation of the facet values found in the search result.

Field Data
Output representation of the field data.

Fields Info
Output representation of the metadata fields in an object.

Index Configuration Collection
Output representation of the collection of index configuration details.

Index Configuration Field
Output representation of the details of the index-configured field.

Index Configurations Update
Output representation of the updated index configuration.

Index Error

Output representation of the error details related to an index.

Index Setting

Output representation of the retrieved index settings.

Index Setting Update

Output representation of the details of the updated index setting.

Invalid Related Object Node

Output representation of the invalid related object node with details of errors.

Metadata

Output representation of the metadata details for objects.

Object Info

Output representation of the object details along with its fields.

Product Classification

Output representation of the product classification details.

Product Component Group

Output representation of the product component group.

Product

Output representation of the product definition.

Product Related Component

Output representation of the product-related component.

Product Selling Model

Output representation of the definition of the product selling model.

Product Selling Model Option

Output representation of the definition of the product selling model option.

Product Specification Type

Output representation of the product specification type.

Products

Output representation of the list of retrieved products.

Related Object Records

Output representation of the related records for a specified record ID and related object API name.

Related Records

Output representation of the list of relatedObject records for a specified record ID.

Related Records List

Output representation of the list of related records.

Search Facet

Output representation of the details of the faceted search.

Setting

Output representation of the setting that's used in indexing.

Setting Metadata

Output representation of the metadata associated with a setting.

Snapshot
Output representation of the list of active snapshots.

Snapshot Collection
Output representation of the retrieved snapshot collection.

Snapshot Deployment
Output representation of the snapshot deployment.

Snapshot Index
Output representation of the snapshot index of a run-time catalog.

Snapshot Index Error
Output representation of the error details related to a snapshot index.

Snapshot Index Info
Output representation of the details of a snapshot index.

Snapshot Index Log
Output representation of a snapshot index log.

Status
Output representation of the status of the request.

Unit of Measure Error
Output representation of the details of errors encountered during the processing of the Unit of Measure API request.

Unit of Measure Info
Output representation of the details of a unit of measure record.

Unit of Measure Status
Output representation of the status of the Unit of Measure API request.

Attribute Category

Output representation of the attribute category.

JSON example

```
"attributeCategory": [
        {
          "attributes": [
            {
              "additionalFields": {
                "scope": "Order"
              },
              "attributeNameOverride": "AD Text",
              "code": "AD02",
              "dataType": "Text",
              "defaultValue": "AD Text DV",
              "description": "AD Text Desc",
              "displayType": "Text",
              "helpText": "AD Text DHT",
              "id": "0tjT1000000002bIAA",
              "isHidden": false,
              "isPriceImpacting": true,
              "isReadOnly": true,
```

```
                "isRequired": true,
                "label": "AD Text Label",
                "maximumCharacterCount": "20",
                "maximumValue": "100",
                "minimumCharacterCount": "1",
                "minimumValue": "50",
                "name": "AD Text",
                "sequence": 1,
                "status": "Active",
                "valueDescription": "AD Text VD"
              }
            ],
            "code": "AC001",
            "id": "0v3T1000000000BIAQ",
            "name": "build and make"
        }
]
```

Property Name	Type	Description	Filter Group and Version	Available Version
attributes	Attribute Definition[]	List of categorized attributes associated with the product.	Small, 60.0	60.0
code	String	Code of the attribute category.	Small, 60.0	60.0
id	String	ID associated with the attribute category.	Small, 60.0	60.0
name	String	Name of the attribute category.	Small, 60.0	60.0

Attribute Definition

Output representation of the attribute definition.

JSON example

```
"attributes": [
     {
      "additionalFields": {
      "scope": "Order"
       },
      "attributeNameOverride": "AD Text",
      "code": "AD02",
      "dataType": "Text",
      "defaultValue": "AD Text DV",
      "description": "AD Text Desc",
      "displayType": "Text",
      "helpText": "AD Text DHT",
      "id": "0tjT1000000002bIAA",
      "isHidden": false,
      "isPriceImpacting": true,
      "isReadOnly": true,
      "isRequired": true,
      "label": "AD Text Label",
```

```
      "maximumCharacterCount": 20,
      "maximumValue": "100",
      "minimumCharacterCount": 1,
      "minimumValue": "50",
      "name": "AD Text",
      "sequence": 1,
      "status": "Active",
      "valueDescription": "AD Text VD"
     }
   ],
    "code": "AC001",
    "id": "0v3T1000000000BIAQ",
    "name": "build and make"
}]
```

Property Name	Type	Description	Filter Group and Version	Available Version
additional Fields	Map<String, Additional Fields Input>	Key-value pair of additional standard or custom fields to include in the response.	Small, 61.0	61.0
attributeName Override	String	Name to display for the attribute, which overrides the name on the attribute. For example, the Color attribute is overridden to display as Laptop Color.	Small, 60.0	60.0
code	String	Unique code of the attribute definition.	Small, 60.0	60.0
dataType	String	Data type of the attribute definition value.	Small, 60.0	60.0
defaultValue	String	Default value of the attribute.	Small, 60.0	60.0
description	String	Description of the attribute.	Small, 60.0	60.0
displayType	String	Display types of the attribute. Valid values are: • Radio Button • Checkbox • Toggle • Input Date • DateTime • Currency Symbol • Currency Code • Currency Name • Percentage • Text • Combobox • Radio Button	Small, 60.0	60.0

Property Name	Type	Description	Filter Group and Version	Available Version
		• `MultiSelect` • `MultiSelectCheckboxes`		
`HelpText`	String	Help text that appears at run time for the attribute.	Small, 60.0	60.0
`id`	String	ID of the attribute definition.	Small, 60.0	60.0
`is Configurable`	Boolean	Reserved for future use.	Small, 60.0	60.0
`isHidden`	Boolean	Indicates whether to hide the attribute from the users in the order capture interface (`true`) or not (`false`).	Small, 60.0	60.0
`isPrice Impacting`	Boolean	Indicates whether the attribute impacts the product price (`true`) or not (`false`).	Small, 60.0	60.0
`isReadOnly`	Boolean	Indicates whether the product attribute is read-only for the end users in the order capture page (`true`) or not (`false`).	Small, 60.0	60.0
`isRequired`	Boolean	Indicates whether a value for the attribute is required for the assigned parent object (`true`) or not (`false`).	Small, 60.0	60.0
`isValue Cloneable`	Boolean	Reserved for future use.	Small, 60.0	60.0
`label`	String	Label of the attribute.	Small, 60.0	60.0
`maximum Character Count`	Integer	Maximum number of alphanumeric characters that can be entered for attributes of type number and text in run time.	Small, 60.0	60.0
`maximumValue`	String	Maximum value that can be entered for attributes of type number, currency, and percent in run time.	Small, 60.0	60.0
`minimum Character Count`	Integer	Minimum number of alphanumeric characters that can be entered for attributes of type number and text in run time. The minimum character count must be less than or equal to the maximum character count.	Small, 60.0	60.0
`minimumValue`	String	Minimum value that can be entered for attributes of type number, currency, and percent in run time. The minimum value must be less than or equal to the maximum value.	Small, 60.0	60.0
`name`	String	Name of the attribute.	Small, 60.0	60.0

Property Name	Type	Description	Filter Group and Version	Available Version
`picklist`	Attribute Picklist	ID of the attribute picklist that provides the valid values for the attribute.	Small, 60.0	60.0
`sequence`	Integer	Order in which the attribute values appear in the attribute definition when the product is configured at run time.	Small, 60.0	60.0
`status`	String	Lifecycle state of the attribute picklist. Valid values are: • `Active` • `Draft` • `Inactive`	Small, 60.0	60.0
`stepValue`	String	Reserved for future use.	Small, 60.0	60.0
`valueDecoder`	String	Reserved for future use.	Small, 60.0	60.0
`value Description`	String	Description of the value assigned to the attribute.	Small, 60.0	60.0

Attribute Picklist

Output representation of the attribute picklist.

JSON example

```
"picklist": {
     "dataType": "Text",
     "description": "Fabric Module options",
     "id": "0v51Q000000TNDkQAO",
     "name": "Fabric Module options",
     "values": [
      {
        "abbreviation": "IFM1"
        "code": "PV0051",
        "displayValue": "25G Intelligent Fabric Module with 8x 25G SFP28 ports",
        "id": "0v61Q0000008OMYQA2",
        "name": "25G Intelligent Fabric Module with 8x 25G SFP28 ports",
        "sequence": "1",
        "value": "25G Intelligent Fabric Module with 8x 25G SFP28 ports",
        "status" : "Active"
      },
      {
        "abbreviation": "IFM2"
        "code": "PV0052",
        "displayValue": "100G Intelligent Fabric Module with 8x 100G QSFP28 ports",
        "id": "0v61Q0000008OMZQA2",
        "name": "100G Intelligent Fabric Module with 8x 100G QSFP28 ports",
        "sequence": "2",
        "value": "100G Intelligent Fabric Module with 8x 100G QSFP28 ports",
```

```
        "status" : "Active"
      }
    ]
  }
```

Property Name	Type	Description	Filter Group and Version	Available Version
dataType	String	Data type of the values in the picklist. Valid values are: • Boolean • Date • Datetime • Number • Text • Currency • Percent	Small, 60.0	60.0
description	String	Description of the picklist, such as the picklist purpose or the associated product.	Small, 60.0	60.0
id	String	ID associated with the attribute picklist record.	Small, 60.0	60.0
name	String	Name of the picklist value.	Small, 60.0	60.0
values	Attribute Picklist Value[]	List of values associated with the picklist.	Small, 60.0	60.0

Attribute Picklist Value

Output representation of the attribute picklist value.

JSON example

```
"values": [
      {
        "abbreviation": "IFM1"
        "code": "PV0051",
        "displayValue": "25G Intelligent Fabric Module with 8x 25G SFP28 ports",
        "id": "0v61Q0000008OMYQA2",
        "name": "25G Intelligent Fabric Module with 8x 25G SFP28 ports",
        "sequence": "1",
        "value": "25G Intelligent Fabric Module with 8x 25G SFP28 ports",
        "status" : "Active"
      }
    ]
```

Property Name	Type	Description	Filter Group and Version	Available Version
abbreviation	String	Name of the picklist value that appears at run time.	Small, 60.0	60.0
code	String	Unique code of the picklist value within the picklist.	Small, 60.0	60.0
displayValue	String	Picklist value that appears at run time in the order capture page.	Small, 60.0	60.0
id	String	ID associated with the attribute picklist value.	Small, 60.0	60.0
name	String	Name of the picklist value.	Small, 60.0	60.0
sequence	String	Order in which the picklist value appears in the picklist.	Small, 60.0	60.0
status	String	Status of the attribute picklist value.	Small, 62.0	62.0
value	String	Value of the picklist item. Value must be unique within the picklist.	Small, 60.0	60.0

Bulk Unit Of Measure Info

Output representation of the details of the unit of measure records along with error details.

JSON example

```
{
  "correlationId": "928ea35f-8a2f-4932-9f7e-ec6cdbcabdbe",
  "errorCodeToErrorMap": {
    "UNIT_OF_MEASURE_INFO_INVALID_UOM_IDS": {
      "errorCode": "UOM_INFO_API_003",
      "messageDetail": "Invalid uomId is passed. Please specify a valid uomId.",
      "messageTitle": "Invalid uomId is passed.",
      "recordIds": [
        "sample"
      ],
      "source": "Unit_Of_Measure_Info_Api"
    }
  },
  "status": {
    "errors": [],
    "httpStatusCode": "200",
    "message": " Successfully fetched UnitOfMeasure Info. "
  },
  "uomIdToUnitOfMeasureInfo": {
    "0hEU200000003M5MAI": {
      "id": "0hEU200000003M5MAI",
      "name": "Pounds",
      "roundingMethod": "Nearest",
      "scale": 1,
```

```
      "unitCode": "Pounds"
    },
    "0hEU200000003KTMAY": {
      "id": "0hEU200000003KTMAY",
      "name": "Grams",
      "roundingMethod": "Down",
      "scale": 5,
      "unitCode": "Grams"
    }
  }
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
correlationId	String	Unique token to track and associate related events or transactions across different components of the application. If unspecified, a Universally Unique Identifier (UUID) is generated.	Small, 63.0	63.0
errorCodeToErrorMap	Map<String, Unit Of Measure Error>	Error codes mapped to their details.	Small, 63.0	63.0
status	Unit Of Measure Status[]	Status of the API request.	Small, 63.0	63.0
uomIdToUnitOfMeasureInfo	Map<String, Unit Of Measure Info>	Unit of measure record IDs mapped to their details.	Small, 63.0	63.0

Data Rounding

Output representation of the data rounding response.

JSON example

```
{
  "keyToUomDataRowOutput": {
    "PRC1": {
      "key": "PRC1",
      "fieldApiNameToFieldDataOutput": {
        "MaxQuantity": {
          "fieldApiName": "MaxQuantity",
          "originalValue": 1234.5678,
          "isRoundingApplicable": true,
          "roundedValue": 1234.56,
          "unitOfMeasureId": "uomId1",
          "errorCodeToErrorMap" : []
        },
        "MinQuantity": {
          "fieldApiName": "MinQuantity",
          "originalValue": 643.1,
          "isRoundingApplicable": true,
          "roundedValue": 643.1,
```

```
          "unitOfMeasureId": "uomId1"
        }
      },
      "errorCodeToErrorMap" : []
    },
    "PRC2": {
      "key": "PRC2",
      "fieldApiNameToFieldDataOutput": {
        "MaxQuantity": {
          "fieldApiName": "MaxQuantity",
          "originalValue": 1234.5678,
          "isRoundingApplicable": true,
          "roundedValue": 1234.56,
          "unitOfMeasureId": "uomId1"
        },
        "MinQuantity": {
          "fieldApiName": "MinQuantity",
          "originalValue": 987.4628,
          "isRoundingApplicable": true,
          "errorCodeToErrorMap": {
            "message": "arithrmetic operation"
          },
          "unitOfMeasureId": "uomId1"
        }
      },
      "errorCodeToErrorMap": []
    },
    "PRC3": {
      "key": "PRC3",
      "fieldApiNameToFieldDataOutput": {
        "MaxQuantity": {
          "fieldApiName": "MaxQuantity",
          "originalValue": 1234.5678,
          "isRoundingApplicable": false,
          "unitOfMeasureId": "uomId2"
        },
        "MinQuantity": {
          "fieldApiName": "MinQuantity",
          "originalValue": 987.4628,
          "isRoundingApplicable": false,
          "unitOfMeasureId": "uomId2"
        }
      },
      "errorCodeToErrorMap": []
    }
  }
}
```


Property Name	Type	Description	Filter Group and Version	Available Version
`correlationId`	String	Unique token to track and associate related events or transactions across different components of the application. If	Small, 63.0	63.0

Property Name	Type	Description	Filter Group and Version	Available Version
		unspecified, a Universally Unique Identifier (UUID) is generated.		
errorCodeToErrorMap	Map<String, Unit Of Measure Error>	Error codes mapped to their details.	Small, 63.0	63.0
keyToDataRowOutput	Map<String, Data Row>	Data row key mapped to the associated data row.	Small, 63.0	63.0
status	Unit Of Measure Status[]	Status of the API request.	Small, 63.0	63.0

Data Row

Output representation of the details of a data row.

JSON example

```
{
  "keyToUomDataRowOutput": {
    "PRC1": {
      "key": "PRC1",
      "fieldApiNameToFieldDataOutput": {
        "MaxQuantity": {
          "fieldApiName": "MaxQuantity",
          "originalValue": 1234.5678,
          "isRoundingApplicable": true,
          "roundedValue": 1234.56,
          "unitOfMeasureId": "uomId1",
          "errorCodeToErrorMap" : []
        },
        "MinQuantity": {
          "fieldApiName": "MinQuantity",
          "originalValue": 643.1,
          "isRoundingApplicable": true,
          "roundedValue": 643.1,
          "unitOfMeasureId": "uomId1"
        }
      },
      "errorCodeToErrorMap" : []
    },
    "PRC2": {
      "key": "PRC2",
      "fieldApiNameToFieldDataOutput": {
        "MaxQuantity": {
          "fieldApiName": "MaxQuantity",
          "originalValue": 1234.5678,
          "isRoundingApplicable": true,
          "roundedValue": 1234.56,
          "unitOfMeasureId": "uomId1"
        },
```

```
        "MinQuantity": {
          "fieldApiName": "MinQuantity",
          "originalValue": 987.4628,
          "isRoundingApplicable": true,
          "errorCodeToErrorMap": {
            "message": "arithrmetic operation"
          },
          "unitOfMeasureId": "uomId1"
        }
      },
      "errorCodeToErrorMap": []
    },
    "PRC3": {
      "key": "PRC3",
      "fieldApiNameToFieldDataOutput": {
        "MaxQuantity": {
          "fieldApiName": "MaxQuantity",
          "originalValue": 1234.5678,
          "isRoundingApplicable": false,
          "unitOfMeasureId": "uomId2"
        },
        "MinQuantity": {
          "fieldApiName": "MinQuantity",
          "originalValue": 987.4628,
          "isRoundingApplicable": false,
          "unitOfMeasureId": "uomId2"
        }
      },
      "errorCodeToErrorMap": []
    }
  }
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
errorCodeToErrorMap	Map<String, Unit Of Measure Error>	Map of error codes to their details.	Small, 63.0	63.0
fieldApiNameToFieldDataOutput	Map<String, Field Data>	Map of field API name to associated field data.	Small, 63.0	63.0
key	String	Unique key of the data row.	Small, 63.0	63.0

Catalog Output

Output representation of the catalog definition.

JSON example

```
catalogs": [
    {
      "catalogType": "Sales",
```

```
      "code": "CAT009",
      "description": "SmartBytes B2B Catalog",
      "effectiveEndDate": "31-07-2023",
      "effectiveStartDate": "24-07-2023",
      "id": "0ZS1Q000000XbZAWA0",
      "name": "SmartBytes B2B Catalog",
      "numberOfCategories": 8
    }
]
```

Property Name	Type	Description	Filter Group and Version	Available Version
catalogType	String	The category of an entry in the catalog, which is customizable. For example, catalog types, such as sellable products, services, parts, technical services, or technical resources.	Small, 60.0	60.0
code	String	Unique ID associated with the catalog.	Small, 60.0	60.0
description	String	Description of the catalog.	Small, 60.0	60.0
effective EndDate	String	Date and time from when the catalog isn't available to the end users.	Small, 60.0	60.0
effective StartDate	String	Date and time from when the catalog is available to the end users.	Small, 60.0	60.0
id	String	ID of the catalog.	Small, 60.0	60.0
name	String	Name of the catalog.	Small, 60.0	60.0
numberOf Categories	Integer	Number of categories in the catalog.	Small, 60.0	60.0

Catalogs Output

Output representation of the retrieved catalog result.

JSON example

```
{
  "catalogs": [
    {
      "catalogType": "Sales",
      "code": "CAT009",
      "id": "0ZS1Q000000XbZAWA0",
      "name": "SmartBytes B2B Catalog",
      "numberOfCategories": 8
    }
  ],
  "correlationId": "0b7b6a30-895c-407a-91b3-e67482d339a3",
  "count": 1,
  "status": {
```

```
    "code": "200",
    "errors": [],
    "message": "Successfully fetched the catalog records."
  }
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`catalogs`	Catalog Output[]	List of the catalogs.	Small, 60.0	60.0
`correlationId`	String	Unique token to track and associate related events or transactions across different components of the application. If unspecified, a Universally Unique Identifier (UUID) is generated.	Small, 60.0	60.0
`count`	Integer	Total number of the catalog records retrieved after the query execution, wherein the `pageSize` property determines the number of records returned in every page.	Small, 60.0	60.0
`status`	Status	Status of the request.	Small, 60.0	60.0

Categories Output

Output representation of the retrieved categories result.

JSON example

```
{
  "categories": [
    {
      "catalogId": "0ZS1Q000000XbZAWA0",
      "code": "B2B Category",
      "description": "Products Category",
      "hasSubCategories": true,
      "id": "0ZG1Q000000XbVGWA0",
      "name": "Unified Computing",
      "numberOfProducts": 2,
      "parentCategoryId": "0ZGT100000000qlOAA",
      "sortOrder": 2,
      "subCategories": [],
      "isNavigational: false
    }
  ],
  "correlationId": "30230973-0a09-405e-b148-f085bb6dd66e",
  "status": {
    "code": "200",
    "errors": [],
    "message": "Successfully fetched the category records."
  }
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
categories	Category Output[]	List of the retrieved categories.	Small, 60.0	60.0
correlationId	String	Unique token to track and associate related events or transactions across different components of the application. If unspecified, a Universally Unique Identifier (UUID) is generated.	Small, 60.0	60.0
status	Status	Status of the request.	Small, 60.0	60.0

Category Output

Output representation of the category definition.

JSON example

```
"categories": [
    {
      "catalogId": "0ZS1Q000000XbZAWA0",
      "code": "B2B Category",
      "description": "Products Category",
      "hasSubCategories": true,
      "id": "0ZG1Q000000XbVGWA0",
      "name": "Unified Computing",
      "numberOfProducts": 2,
      "parentCategoryId": "0ZGT100000000qlOAA",
      "sortOrder": 2,
      "subCategories": [],
      "isNavigational: false
    }
  ]
```

Property Name	Type	Description	Filter Group and Version	Available Version
catalogId	String	ID of the catalog that the category is associated with.	Small, 60.0	60.0
code	String	Unique code of the product category.	Small, 60.0	60.0
description	String	Description of the category. If data translation is set up and specified in the org, the translated description is available.	Small, 60.0	60.0
hasSub Categories	Boolean	Indicates whether the subcategories are available (true) or not (false).	Small, 60.0	60.0
id	String	ID of the category.	Small, 60.0	60.0
isNavigational	Boolean	Indicates whether the category node is navigational (`true`) or not (`false`).	Small, 62.0	62.0

Property Name	Type	Description	Filter Group and Version	Available Version
name	String	Name of the category. If data translation is set up and specified in the org, the translated name is available.	Small, 60.0	60.0
numberOf Products	Integer	Number of products associated with the category.	Small, 60.0	60.0
parent CategoryId	String	ID of the parent category.	Small, 60.0	60.0
sortOrder	Integer	Display order of the product category relative to the siblings with the same parent category.	Small, 60.0	60.0
subCategories	Category Output[]	List of subcategories, if available. This property is returned with the Categories List (GET) API response.	Small, 60.0	60.0

Deep Clone Error

Output representation of the error details related to the deep clone request.

Property Name	Type	Description	Filter Group and Version	Available Version
errorCode	String	Code of the error message related to the deep clone request.	Small, 63.0	63.0
errorMessage	String	Details of the error message related to the deep clone request.	Small, 63.0	63.0

Deep Clone Record Response

Output representation of the details of the cloned related records.

JSON example

```
"createdRecordList": [
  {
    "createdRecordId": "01tSG0000030Yb3YAE",
    "entityApiName": "Product2",
    "entityLabel": "Product"
  },
  {
    "createdRecordId": "0iOSG0000002rMn2AI",
    "entityApiName": "ProductSellingModelOption",
    "entityLabel": "Product Selling Model Option"
  },
  {
    "createdRecordId": "0v7SG0000001ktdYAA",
```

```
      "entityApiName": "ProductAttributeDefinition",
      "entityLabel": "Product Attribute Definition"
    }
  ]
```

Property Name	Type	Description	Filter Group and Version	Available Version
created RecordId	String	ID of the created related record.	Small, 63.0	63.0
entityApiName	String	API name of the created object.	Small, 63.0	63.0
entityLabel	String	Label of the created object.	Small, 63.0	63.0

Deep Clone Response

Output representation of the details of the cloned record.

JSON example

```
{
  "createdRecordList": [
    {
      "createdRecordId": "01tSG0000030Yb3YAE",
      "entityApiName": "Product2",
      "entityLabel": "Product"
    },
    {
      "createdRecordId": "0iOSG0000002rMn2AI",
      "entityApiName": "ProductSellingModelOption",
      "entityLabel": "Product Selling Model Option"
    },
    {
      "createdRecordId": "0v7SG0000001ktdYAA",
      "entityApiName": "ProductAttributeDefinition",
      "entityLabel": "Product Attribute Definition"
    }
  ],
  "createdRootRecordId": "01tSG0000030Yb3YAE",
  "errorList": [],
  "isSuccessful": true
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
created RecordList	Deep Clone Record Response[]	List of cloned related records of the main record.	Small, 63.0	63.0
createdRoot RecordId	String	ID of the created root record.	Small, 63.0	63.0
errorList	Deep Clone Error[]	Details of errors, if any.	Small, 63.0	63.0

Property Name	Type	Description	Filter Group and Version	Available Version
errorMessage	String	Error message if the API request fails.	Small, 63.0	63.0
isSuccessful	Boolean	Indicates whether the API request is successful (true) or not (false).	Small, 63.0	63.0

Error Output

Output representation of the error details.

Property Name	Type	Description	Filter Group and Version	Available Version
errorCode	String	Code of the error message.	Small, 60.0	60.0
messageDetail	String	Details of the error message.	Small, 60.0	60.0
messageTitle	String	Title of the error message.	Small, 60.0	60.0
nodeProductId	String	ID of the product node.	Small, 61.0	61.0
recordId	String	ID of the record.	Small, 60.0	60.0
recordName	String	Name of the record.	Small, 60.0	60.0
relatedObjectNodes	Invalid Related Object Node[]	List of related object nodes with errors.	Small, 62.0	62.0
source	String	Name of the API that's the source of the error.	Small, 60.0	60.0

Facet Value

Output representation of the facet values found in the search result.

JSON example

```
"values": [
  {
    "displayName": "Simple",
    "nameOrId": "Simple",
    "productCount": 9
  }
]
```

Property Name	Type	Description	Filter Group and Version	Available Version
displayName	String	Display name of the facet value.	Small, 63.0	63.0
nameOrId	String	Facet value name or ID. Reserved for internal use.	Small, 63.0	63.0

Property Name	Type	Description	Filter Group and Version	Available Version
productCount		Number of products in the search result that match the facet value.	Small, 63.0	63.0

Field Data

Output representation of the field data.

Property Name	Type	Description	Filter Group and Version	Available Version
errorCodeToErrorMap	Map<String, Unit Of Measure Error>	Error codes mapped to their details.	Small, 63.0	63.0
fieldApiName	String	Unique API Name of the field.	Small, 63.0	63.0
isRoundingApplicable	Boolean	Indicates whether data rounding is applicable to the decimal (true) or not (false).	Small, 63.0	63.0
originalValue	String	Original value of the field.	Small, 63.0	63.0
roundedValue	String	Rounded field value that corresponds to the original value, if data rounding is applicable.	Small, 63.0	63.0
unitOfMeasureId	String	ID of the unit of measure record that's associated to the field.	Small, 63.0	63.0

Fields Info

Output representation of the metadata fields in an object.

JSON example

```
"fields": [
  {
    "dataType": "text",
    "isFacetableConfigurable": true,
    "isSearchableConfigurable": false,
    "label": "Product Name",
    "name": "Name",
    "type": "Standard"
  },
  {
    "dataType": "multilinetext",
    "isFacetableConfigurable": false,
    "isSearchableConfigurable": true,
    "label": "Product Description",
    "name": "Description",
    "type": "Standard"
  }
]
```

Property Name	Type	Description	Filter Group and Version	Available Version
customFieldId	String	ID of the custom field.	Small, 62.0	62.0
dataType	String	Type of data.	Small, 62.0	62.0
isConfigurable	Boolean	Reserved for internal use.	Small, 62.0	62.0
isFacetableConfigurable	Boolean	Indicates whether the field is facetable (true) or not (false).	Small, 63.0	Small, 63.0
isSearchableConfigurable	Boolean	Indicates whether the field is searchable (true) or not (false).	Small, 63.0	Small, 63.0
label	String	Label of the object field.	Small, 62.0	62.0
name	String	Name of the object field.	Small, 62.0	62.0
type	String	Type of the object field.	Small, 62.0	62.0

Index Configuration Collection

Output representation of the collection of index configuration details.

JSON example

```
{
  "correlationId": "ad960cb6-392d-4d11-bac3-3824baedf67e",
  "errors": [],
  "indexConfigurations": [
    {
      "isSearchable": true,
      "name": "Name",
      "type": "Standard"
    }
  ],
  "metadata": {
    "objectInfos": [
      {
        "fields": [
          {
            "dataType": "text",
            "isFacetableConfigurable": true,
            "isSearchableConfigurable": false,
            "label": "Product Name",
            "name": "Name",
            "type": "Standard"
          },
          {
            "dataType": "multilinetext",
            "isFacetableConfigurable": false,
            "isSearchableConfigurable": true,
            "label": "Product Description",
```

```
            "name": "Description",
            "type": "Standard"
          }
        ],
        "name": "Product2"
      },
      {
        "fields": [
          {
            "dataType": "stringplusclob",
            "label": "Description",
            "name": "Description",
            "type": "ProductAttributeDefinitionStandard"
          },
          {
            "dataType": "text",
            "label": "Name",
            "name": "Name",
            "type": "ProductAttributeDefinitionStandard"
          },
        ],
        "name": "ProductAttributeDefinition"
      }
    ]
  },
  "statusCode": "200"
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
correlationId	String	Unique token to track and associate related events or transactions across different components of the application. If unspecified, a Universally Unique Identifier (UUID) is generated.	Small, 62.0	62.0
errors	Error Output[]	List of errors, if any.	Small, 62.0	62.0
indexConfigurations	Index Configuration Field[]	Details of the index-configured fields.	Small, 62.0	62.0
metadata	Metadata[]	Details of the metadata for objects.	Small, 62.0	62.0
statusCode	String	Code that indicates the status of the request.	Small, 62.0	62.0

Index Configuration Field

Output representation of the details of the index-configured field.

JSON example

```
  "indexConfigurations": [
    {
```

```
      "attributeDefinitionId": "0tjT1000000002bIAA",
      "name": "Color",
      "type": "ProductDynamicAttribute",
      "isSearchable": true
    },
    {
      "attributeFieldId": "00Nxx000001FwnABII",
      "name": "Message__c",
      "type": "Custom",
      "isSearchable": true
    },
    {
      "name": "Code",
      "type": "Standard",
      "isSearchable": true
    },
    {
      "facetDisplayRank": 1,
      "isFacetable": false,
      "isSearchable": true,
      "name": "Family",
      "type": "Standard"
    }
  ]
```

Property Name	Type	Description	Filter Group and Version	Available Version
`attributeDefinitionId`	String	ID of the attribute definition.	Small, 62.0	62.0
`attributeFieldId`	String	ID of the attribute field.	Small, 62.0	62.0
`facetDisplayRank`	Integer	Sort order for displaying the facets at run time.	Small, 63.0	63.0
`isFacetable`	Boolean	Indicates whether the field is facetable (`true`) or not (`false`).	Small, 63.0	63.0
`isSearchable`	Boolean	Indicates whether the index-configured field is searchable (`true`) or not (`false`).	Small, 62.0	62.0
`name`	String	Name of the index-configured field.	Small, 62.0	62.0
`type`	String	Type of the index-configured field.	Small, 62.0	62.0

Index Configurations Update

Output representation of the updated index configuration.

JSON example

```
{
  "correlationId": "8545b5aa-f3e6-429a-8f21-9cc4ce50b1d7",
```

```
  "errors": [],
  "indexConfigurations": [
    {
      "attributeDefinitionId": "0tjT1000000002bIAA",
      "name": "Color",
      "type": "ProductDynamicAttribute",
      "isSearchable": true
    },
    {
      "attributeFieldId": "00Nxx000001FwnABII",
      "name": "Message__c",
      "type": "Custom",
      "isSearchable": true
    },
    {
      "name": "Code",
      "type": "Standard",
      "isSearchable": true
    },
    {
      "facetDisplayRank": 1,
      "isFacetable": false,
      "isSearchable": true,
      "name": "Family",
      "type": "Standard"
    }
  ],
  "statusCode": "200"
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
correlationId	String	Unique token to track and associate related events or transactions across different components of the application. If unspecified, a Universally Unique Identifier (UUID) is generated.	Small, 62.0	62.0
errors	Error Output[]	List of errors, if any.	Small, 62.0	62.0
indexConfigurations	Index Configuration Field[]	Details of the index-configured fields.	Small, 62.0	62.0
statusCode	String	Code that indicates the status of the request.	Small, 62.0	62.0

Index Error

Output representation of the error details related to an index.

JSON example

```
  "indexErrorDetails": {
    "errorFileId": "069xx0000004C92AAE",
```

```
    "indexCreatedDate": "2024-10-03T05:24:18.000Z",
    "indexErrorsCount": 1,
    "indexLastUpdatedDate": "2024-10-03T05:27:00.000Z",
    "itemLevelErrorsCount": 1
  }
```

Property Name	Type	Description	Filter Group and Version	Available Version
errorFileId	String	ID of the exported error file that contains the index errors.	Small, 63.0	63.0
indexCreatedDate	String	Date on which the index was created.	Small, 63.0	63.0
indexErrorsCount	Integer	Number of index-level errors.	Small, 63.0	63.0
indexLastUpdatedDate	String	Date on which the index was last updated.	Small, 63.0	63.0
itemLevelErrorsCount	Integer	Number of item-level errors.	Small, 63.0	63.0

Index Setting

Output representation of the retrieved index settings.

JSON example

```
{
  "errors": [],
  "metadata": {
    "activeLanguages": ["en_US","ja","es","nl_NL"]
  },
  "setting": {
    "defaultLanguage": "en_US",
    "id": "1JySG0000000GUb0AM",
    "supportedLanguages": ["en_US","ja","es","nl_NL"]
  },
  "statusCode": "200"
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
errors	Error Output[]	List of errors, if any.	Small, 63.0	63.0
metadata	Setting Metadata[]	Metadata associated with the setting.	Small, 63.0	63.0
setting	Setting[]	Setting that's used in indexing and maintained for an org.	Small, 63.0	63.0
statusCode	String	Code that indicates the status of the request.	Small, 63.0	63.0

Index Setting Update

Output representation of the details of the updated index setting.

JSON example

```
{
   "setting" : {
        "supportedLanguages" : ["en_US","ja","es","nl_NL"],
        "id": "1JySG0000000GUb0AM",
        "defaultLanguage" : "en_US"
   },
   "errors" : [],
   "statusCode" : "200"
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
errors	Error Output[]	List of errors, if any.	Small, 63.0	63.0
setting	Setting[]	Setting that's used in indexing and maintained for an org.	Small, 63.0	63.0
statusCode	String	Code that indicates the status of the API request.	Small, 63.0	63.0

Invalid Related Object Node

Output representation of the invalid related object node with details of errors.

JSON example

```
To add
```

Property Name	Type	Description	Filter Group and Version	Available Version
errorMessages	String[]	List of error messages.	Small, 62.0	62.0
relatedObjectAPIName	String	API name of the related object.	Small, 62.0	62.0

Metadata

Output representation of the metadata details for objects.

JSON example

```
  "metadata": {
    "objectInfos": [
      {
        "fields": [
          {
            "dataType": "text",
```

```
            "isFacetableConfigurable": true,
            "isSearchableConfigurable": false,
            "label": "Product Name",
            "name": "Name",
            "type": "Standard"
          },
          {
            "dataType": "multilinetext",
            "isFacetableConfigurable": false,
            "isSearchableConfigurable": true,
            "label": "Product Description",
            "name": "Description",
            "type": "Standard"
          }
        ],
        "name": "Product2"
      },
      {
        "fields": [
          {
            "dataType": "stringplusclob",
            "label": "Description",
            "name": "Description",
            "type": "ProductAttributeDefinitionStandard"
          },
          {
            "dataType": "text",
            "label": "Name",
            "name": "Name",
            "type": "ProductAttributeDefinitionStandard"
          },
        ],
        "name": "ProductAttributeDefinition"
      }
    ]
  }
```

Property Name	Type	Description	Filter Group and Version	Available Version
objectInfos	Object Info[]	Metadata details for objects.	Small, 62.0	62.0

Object Info

Output representation of the object details along with its fields.

JSON example

```
  "objectInfos": [
    {
      "fields": [
        {
          "dataType": "text",
          "isFacetableConfigurable": true,
```

```
            "isSearchableConfigurable": false,
            "label": "Product Name",
            "name": "Name",
            "type": "Standard"
          },
          {
            "dataType": "multilinetext",
            "isFacetableConfigurable": false,
            "isSearchableConfigurable": true,
            "label": "Product Description",
            "name": "Description",
            "type": "Standard"
          }
        ],
        "name": "Product2"
      },
      {
        "fields": [
          {
            "dataType": "stringplusclob",
            "label": "Description",
            "name": "Description",
            "type": "ProductAttributeDefinitionStandard"
          },
          {
            "dataType": "text",
            "label": "Name",
            "name": "Name",
            "type": "ProductAttributeDefinitionStandard"
          },
        ],
        "name": "ProductAttributeDefinition"
      }
    ]
```

Property Name	Type	Description	Filter Group and Version	Available Version
fields	Fields Info[]	Fields of the object.	Small, 62.0	62.0
name	String	Name of the object.	Small, 62.0	62.0

Product Classification

Output representation of the product classification details.

JSON example

```
"productClassification": {
              "id": "11BT10000004C9SMAU",
              "name" : "class",
              "code" : "code",
              "status" : "Active"
            }
```

Property Name	Type	Description	Filter Group and Version	Available Version
code	String	Code of the product classification record.	Small, 61.0	61.0
id	String	ID of the product classification record.	Small, 60.0	60.0
name	String	Name of the product classification record.	Small, 61.0	61.0
status	String	Status of the product classification record.	Small, 61.0	61.0

Product Component Group

Output representation of the product component group.

JSON example


```
"productComponentGroups":[
  {
    "childGroups":[
     {
      "childGroups":[],
    "    components":[
          {
            "additionalFields":{},
            "attributeCategory":[],
            "attributes":[],
            "catalogs":[],
            "categories":[],
            "childProducts":[],
            "id":"01txx0000006i2aAAA",
            "isActive":true,
            "isAssetizable":true,
            "isSoldOnlyWithOtherProds":true,
            "name":"GenWatt Diesel 1000kW",
            "nodeType":"simpleProduct",
            "productCode":"GC1060",
            "productComponentGroups":[],
            "productRelatedComponent":
             {
              "childProductId":"01txx0000006i2aAAA",
              "doesBundlePriceIncludeChild":true,
              "id":"0dSxx00000001P7EAI",
              "isComponentRequired":false,
              "isDefaultComponent":true,
              "isExcluded":false,
              "isQuantityEditable":false,
              "parentProductId":"01txx0000006iC8AAI",
              "productRelationshipTypeId":"0yoxx000000001dAAA",
              "quantity":1,
              "quantityScaleMethod":"Proportional"
              },
              "productSellingModelOptions":[]
              },
              {
```

```
"additionalFields":{},
"attributeCategory":[],
"attributes":[],
"catalogs":[],
"categories":[],
"childProducts":[],
"id":"01txx0000006i2TAAQ",
"isActive":true,
"isAssetizable":true,
"isSoldOnlyWithOtherProds":true,
"name":"GenWatt Diesel 10kW",
"nodeType":"simpleProduct",
"productCode":"GC1020",
"productComponentGroups":[],
"productRelatedComponent":{
    "childProductId":"01txx0000006i2TAAQ",
    "doesBundlePriceIncludeChild":true,
    "id":"0dSxx00000001P8EAI",
    "isComponentRequired":false,
    "isDefaultComponent":true,
    "isExcluded":false,
    "isQuantityEditable":false,
    "parentProductId":"01txx0000006iC8AAI",
    "productRelationshipTypeId":"0yoxx000000001dAAA",
    "quantity":1,"quantityScaleMethod":"Proportional"
 },
"productSellingModelOptions":[]
 },
 {
  "additionalFields":{},
  "attributeCategory":[],
  "attributes":[],
  "catalogs":[],
  "categories":[],
  "childProducts":[],
  "id":"01txx0000006i2SAAQ",
  "isActive":true,
  "isAssetizable":true,
  "isSoldOnlyWithOtherProds":true,
  "name":"GenWatt Diesel 200kW",
  "nodeType":"simpleProduct",
  "productCode":"GC1040",
  "productComponentGroups":[],
  "productRelatedComponent":
    {
      "childProductId":"01txx0000006i2SAAQ",
      "doesBundlePriceIncludeChild":true,
      "id":"0dSxx00000001P9EAI",
      "isComponentRequired":false,
      "isDefaultComponent":true,
      "isExcluded":false,
      "isQuantityEditable":false,
      "parentProductId":"01txx0000006iC8AAI",
      "productRelationshipTypeId":"0yoxx000000001dAAA",
```

```
                "quantity":1,
                "quantityScaleMethod":"Proportional"
               },
              "productSellingModelOptions":[]
                }
              ],
              "id":"0y7xx0000000151AAA",
              "isExcluded":false,
              "name":"G1.1",
              "parentGroupId":"0y7xx0000000149AAA",
              "parentProductId":"01txx0000006iC8AAI"
               }
              ],
            "components":[],
            "id":"0y7xx0000000149AAA",
            "isExcluded":false,
            "name":"G1",
            "parentProductId":"01txx0000006iC8AAI"
            }
        ]
```


Property Name	Type	Description	Filter Group and Version	Available Version
`code`	String	Unique code of the product component group, which is used only during design time.	Small, 60.0	60.0
`components`	Product[]	List of the product details.	Small, 60.0	60.0
`childGroups`	Product Component Group[]	List of child product components groups.	Small, 62.0	62.0
`description`	String	Description of the product component group.	Small, 60.0	60.0
`id`	String	ID of the record.	Small, 60.0	60.0
`isExcluded`	Boolean	Indicates whether the product component group is excluded from the product bundle for selection in the run time (`true`) or not (`false`).	Small, 60.0	60.0
`maxBundleComponents`	Integer	Maximum number of product components that can be added to a group.	Small, 60.0	60.0
`minBundleComponents`	Integer	Minimum number of product components that can be added to a group.	Small, 60.0	60.0
`name`	String	Name of the record.	Small, 60.0	60.0
`parentProductId`	String	ID associated with the parent product record.	Small, 60.0	60.0
`parentGroupId`	String	ID of the parent group record.	Small, 62.0	62.0

Property Name	Type	Description	Filter Group and Version	Available Version
sequence	Integer	Order in which the groups are listed in the bundle.	Small, 60.0	60.0

Product

Output representation of the product definition.

JSON example

This example shows a sample of the Product List (POST) API response.

```
{
  "products": [
      {
      "additionalFields": {
        "code__c": "SWX445"
      },
       "attributeCategory": [
        {
          "attributes": [
            {
              "additionalFields": {
                "scope": "Order"
              },
              "attributeNameOverride": "AD Text",
              "code": "AD02",
              "dataType": "Text",
              "displayType": "Text",
              "MinimumCharacterCount": "1",
              "MaximumCharacterCount": "20",
              "defaultValue": "AD Text DV",
              "description": "AD Text Desc",
              "helpText": "AD Text DHT",
              "id": "0tjT1000000002bIAA",
              "isHidden": false,
              "isPriceImpacting": true,
              "isReadOnly": true,
              "isRequired": true,
              "label": "AD Text Label",
              "name": "AD Text",
              "sequence": 1,
              "status": "Active",
              "valueDescription": "AD Text VD"
            }
          ],
          "code": "AC001",
          "id": "0v3T1000000000BIAQ",
          "name": "build and make"
        }
      ],
      "attributes": [
        {
```

```
"additionalFields": {
  "scope": "SWX445"
},
"attributeNameOverride": "AD Picklist",
"code": "AD001",
"dataType": "Picklist",
"defaultValue": "Red",
"description": "AD Picklist Description",
"helpText": "AD Picklist DHT",
"id": "0tjT1000000002WIAQ",
"isHidden": false,
"isPriceImpacting": false,
"isReadOnly": false,
"isRequired": true,
"label": "AD Picklist Label",
"name": "AD Picklist",
"picklist": {
  "dataType": "Text",
  "description": "APV Description",
  "id": "0v5T10000000001IAA",
  "name": "Color",
  "values": [
    {
      "abbreviation": "Blue Abb",
      "code": "APV03",
      "displayValue": "Blue DV",
      "id": "0v6T10000000006IAA",
      "name": "Blue",
      "sequence": "3",
      "value": "Blue b",
      "status": "Active"
    },
    {
      "abbreviation": "Red Abb",
      "code": "APV04",
      "displayValue": "Red",
      "id": "0v6T10000000001IAA",
      "name": "Red",
      "sequence": "4",
      "value": "Red",
      "status": "Active"
    },
    {
      "abbreviation": "One Abb",
      "code": "APV02",
      "displayValue": "One DV",
      "id": "0v6T1000000000uIAA",
      "name": "One",
      "sequence": "2",
      "value": "One 1",
      "status": "Active"
    },
    {
      "abbreviation": "Red Abbreviation",
```

```
                "code": "APV01",
                "displayValue": "Red Display Value",
                "id": "0v6T1000000001OIAQ",
                "name": "Red",
                "sequence": "1",
                "value": "red12",
                "status": "Active"
              }
            ]
          },
          "sequence": 1,
          "status": "Active",
          "valueDescription": "AD Picklist VD"
        }
      ],
      "categories": [],
      "childProducts": [
        {
          "attributeCategory": [],
          "attributes": [],
          "categories": [],
          "childProducts": [],
          "configureDuringSale": "NotAllowed",
          "id": "01tZ7000000AJkaIAG",
          "isActive": false,
          "isAssetizable": true,
          "isSoldOnlyWithOtherProds": false,
          "name": "Earphones",
          "nodeType": "bundleProduct",
          "productComponentGroups": [],
          "productRelatedComponent": {
            "childProductId": "01tZ7000000AJkaIAG",
            "doesBundlePriceIncludeChild": true,
            "id": "0dSZ700000000cdMAA",
            "isComponentRequired": false,
            "isDefaultComponent": true,
            "isExcluded": false,
            "isQuantityEditable": false,
            "parentProductId": "01tZ7000000AJXOIA4",
            "productRelationshipTypeId": "0yoZ700000000kPIAQ",
            "quantity": 1,
            "quantityScaleMethod": "Proportional"
          },
          "productSellingModelOptions": []
        }
      ],
     "description": "Keep your organization connected with seamless collaboration across
 distributed teams. No matter where employees are located, organizations are seeking
stronger employee engagement and customer experiences to enable more productivity and
greater business agility. More effective collaboration helps organizations work smarter.",

      "displayUrl":
"https://dispatch.m.io/wp-content/uploads/2023/01/History-of-Webex.png",
      "id": "01t1Q000008CD2eQAG",
```

```
    "isActive": true,
    "isAssetizable": true,
    "isSoldOnlyWithOtherProds": false,
    "name": "SmartBytes Collaboration Suite",
    "nodeType": "bundleProduct",
    "productClassification": {
      "id": "11B1Q0000008OMGUA2"
    },
    "productCode": "P0143",
    "productComponentGroups": [
      {
        "components": [
          {
            "attributeCategory": [],
            "attributes": [],
            "categories": [],
            "childProducts": [],
            "id": "01tZ7000000AJXTIA4",
            "isActive": false,
            "isAssetizable": true,
            "isSoldOnlyWithOtherProds": false,
            "name": "Charger",
            "nodeType": "bundleProduct",
            "productComponentGroups": [],
            "productRelatedComponent": {
              "childProductId": "01tZ7000000AJXTIA4",
              "doesBundlePriceIncludeChild": true,
              "id": "0dSZ700000000YLMAY",
              "isComponentRequired": false,
              "isDefaultComponent": true,
              "isExcluded": false,
              "isQuantityEditable": false,
              "parentProductId": "01tZ7000000AJXOIA4",
              "productRelationshipTypeId": "0yoZ700000000kPIAQ",
              "quantity": 1,
              "quantityScaleMethod": "Proportional"
            },
            "productSellingModelOptions": []
          }
        ],
        "id": "0y7Z700000000TtIAI",
        "isExcluded": false,
        "name": "Box",
        "parentProductId": "01tZ7000000AJXOIA4"
      }
    ],
    "productSellingModelOptions": [
      {
        "id": "0iO1Q0000008OkeUAE",
        "productId": "01t1Q000008CD2eQAG",
        "productSellingModel": {
          "id": "0jP1Q000000CaVFUA0",
          "isDefault": true,
          "name": "One Time",
```

```
            "sellingModelType": "OneTime",
            "status": "Active"
          }
        }
      ],
      "productSpecificationType": {
        "name": "None"
      }
    }
],
  "status": {
    "code": "200",
    "correlationId": "fd158d80-d73c-4a1f-a009-9225db804d70",
    "errors": [],
    "message": "Successfully fetched product records."
  }
}
```


This example shows a sample of the Bulk Product Details (POST) API response.


```
{
  "products": [
      {
      "additionalFields": {
        "code__c": "SWX445"
      },
       "attributeCategory": [
        {
          "attributes": [
            {
              "additionalFields": {
                "scope": "Order"
              },
              "attributeNameOverride": "AD Text",
              "code": "AD02",
              "dataType": "Text",
              "displayType": "Text",
              "MinimumCharacterCount": "1",
              "MaximumCharacterCount": "20",
              "defaultValue": "AD Text DV",
              "description": "AD Text Desc",
              "helpText": "AD Text DHT",
              "id": "0tjT1000000002bIAA",
              "isHidden": false,
              "isPriceImpacting": true,
              "isReadOnly": true,
              "isRequired": true,
              "label": "AD Text Label",
              "name": "AD Text",
              "sequence": 1,
              "status": "Active",
              "valueDescription": "AD Text VD"
            }
          ],
          "code": "AC001",
```

```
        "id": "0v3T1000000000BIAQ",
        "name": "build and make"
      }
    ],
    "attributes": [
      {
        "additionalFields": {
          "scope": "SWX445"
        },
        "attributeNameOverride": "AD Picklist",
        "code": "AD001",
        "dataType": "Picklist",
        "defaultValue": "Red",
        "description": "AD Picklist Description",
        "helpText": "AD Picklist DHT",
        "id": "0tjT1000000002WIAQ",
        "isHidden": false,
        "isPriceImpacting": false,
        "isReadOnly": false,
        "isRequired": true,
        "label": "AD Picklist Label",
        "name": "AD Picklist",
        "picklist": {
          "dataType": "Text",
          "description": "APV Description",
          "id": "0v5T10000000001IAA",
          "name": "Color",
          "values": [
            {
              "abbreviation": "Blue Abb",
              "code": "APV03",
              "displayValue": "Blue DV",
              "id": "0v6T10000000006IAA",
              "name": "Blue",
              "sequence": "3",
              "value": "Blue b",
              "status": "Active"
            },
            {
              "abbreviation": "Red Abb",
              "code": "APV04",
              "displayValue": "Red",
              "id": "0v6T10000000001IAA",
              "name": "Red",
              "sequence": "4",
              "value": "Red",
              "status": "Active"
            },
            {
              "abbreviation": "One Abb",
              "code": "APV02",
              "displayValue": "One DV",
              "id": "0v6T1000000000uIAA",
              "name": "One",
```

```
                "sequence": "2",
                "value": "One 1",
                "status": "Active"
              },
              {
                "abbreviation": "Red Abbreviation",
                "code": "APV01",
                "displayValue": "Red Display Value",
                "id": "0v6T1000000001OIAQ",
                "name": "Red",
                "sequence": "1",
                "value": "red12",
                "status": "Active"
              }
            ]
          },
          "sequence": 1,
          "status": "Active",
          "valueDescription": "AD Picklist VD"
        }
      ],
      "categories": [],
      "childProducts": [
        {
          "attributeCategory": [],
          "attributes": [],
          "categories": [],
          "childProducts": [],
          "configureDuringSale": "NotAllowed",
          "id": "01tZ7000000AJkaIAG",
          "isActive": false,
          "isAssetizable": true,
          "isSoldOnlyWithOtherProds": false,
          "name": "Earphones",
          "nodeType": "bundleProduct",
          "productComponentGroups": [],
          "productRelatedComponent": {
            "childProductId": "01tZ7000000AJkaIAG",
            "doesBundlePriceIncludeChild": true,
            "id": "0dSZ700000000cdMAA",
            "isComponentRequired": false,
            "isDefaultComponent": true,
            "isExcluded": false,
            "isQuantityEditable": false,
            "parentProductId": "01tZ7000000AJXOIA4",
            "productRelationshipTypeId": "0yoZ700000000kPIAQ",
            "quantity": 1,
            "quantityScaleMethod": "Proportional"
          },
          "productSellingModelOptions": []
        }
      ],
     "description": "Keep your organization connected with seamless collaboration across
distributed teams. No matter where employees are located, organizations are seeking
```

```
stronger employee engagement and customer experiences to enable more productivity and
greater business agility. More effective collaboration helps organizations work smarter.",

      "displayUrl":
"https://dispatch.m.io/wp-content/uploads/2023/01/History-of-Webex.png",
      "id": "01t1Q000008CD2eQAG",
      "isActive": true,
      "isAssetizable": true,
      "isSoldOnlyWithOtherProds": false,
      "name": "SmartBytes Collaboration Suite",
      "nodeType": "bundleProduct",
      "productClassification": {
        "id": "11B1Q0000008OMGUA2",
        "name" : "class",
        "code" : "code",
        "status" : "Active"
      },
      "productCode": "P0143",
      "productComponentGroups": [
        {
          "components": [
            {
              "attributeCategory": [],
              "attributes": [],
              "categories": [],
              "childProducts": [],
              "id": "01tZ7000000AJXTIA4",
              "isActive": false,
              "isAssetizable": true,
              "isSoldOnlyWithOtherProds": false,
              "name": "Charger",
              "nodeType": "bundleProduct",
              "productComponentGroups": [],
              "productRelatedComponent": {
                "childProductId": "01tZ7000000AJXTIA4",
                "doesBundlePriceIncludeChild": true,
                "id": "0dSZ700000000YLMAY",
                "isComponentRequired": false,
                "isDefaultComponent": true,
                "isExcluded": false,
                "isQuantityEditable": false,
                "parentProductId": "01tZ7000000AJXOIA4",
                "productRelationshipTypeId": "0yoZ700000000kPIAQ",
                "quantity": 1,
                "quantityScaleMethod": "Proportional"
              },
              "productSellingModelOptions": []
            }
          ],
          "id": "0y7Z700000000TtIAI",
          "isExcluded": false,
          "name": "Box",
          "parentProductId": "01tZ7000000AJXOIA4"
        }
```

```
      ],
      "productSellingModelOptions": [
        {
          "id": "0iO1Q0000008OkeUAE",
          "productId": "01t1Q000008CD2eQAG",
          "productSellingModel": {
            "id": "0jP1Q000000CaVFUA0",
            "isDefault": true,
            "name": "One Time",
            "sellingModelType": "OneTime",
            "status": "Active"
          }
        }
      ],
      "productSpecificationType": {
        "name": "None"
      }
    }
],
  "status": {
    "code": "200",
    "correlationId": "fd158d80-d73c-4a1f-a009-9225db804d70",
    "errors": [],
    "message": "Successfully fetched product records."
  }
}
```


This example shows a sample of the Product By ID (GET) API response.


```
{
  "products": [
    {
      "attributeCategory": [
        {
          "attributes": [
            {
              "attributeNameOverride": "AD Text",
              "code": "AD02",
              "dataType": "Text",
              "displayType": "Text",
              "MinimumCharacterCount": "1",
              "MaximumCharacterCount": "20",
              "defaultValue": "AD Text DV",
              "description": "AD Text Desc",
              "helpText": "AD Text DHT",
              "id": "0tjT1000000002bIAA",
              "isHidden": false,
              "isPriceImpacting": true,
              "isReadOnly": true,
              "isRequired": true,
              "label": "AD Text Label",
              "name": "AD Text",
              "sequence": 1,
              "status": "Active",
              "valueDescription": "AD Text VD"
```

```
            }
          ],
          "code": "AC001",
          "id": "0v3T1000000000BIAQ",
          "name": "build and make"
        }
      ],
      "attributes": [
        {
          "attributeNameOverride": "AD Picklist",
          "code": "AD001",
          "dataType": "Picklist",
          "defaultValue": "Red",
          "description": "AD Picklist Description",
          "helpText": "AD Picklist DHT",
          "id": "0tjT1000000002WIAQ",
          "isHidden": false,
          "isPriceImpacting": false,
          "isReadOnly": false,
          "isRequired": true,
          "label": "AD Picklist Label",
          "name": "AD Picklist",
          "picklist": {
            "dataType": "Text",
            "description": "APV Description",
            "id": "0v5T10000000001IAA",
            "name": "Color",
            "values": [
              {
                "abbreviation": "Blue Abb",
                "code": "APV03",
                "displayValue": "Blue DV",
                "id": "0v6T10000000006IAA",
                "name": "Blue",
                "sequence": "3",
                "value": "Blue b",
                "status": "Active"
              },
              {
                "abbreviation": "Red Abb",
                "code": "APV04",
                "displayValue": "Red",
                "id": "0v6T10000000001IAA",
                "name": "Red",
                "sequence": "4",
                "value": "Red",
                "status": "Active"
              },
              {
                "abbreviation": "One Abb",
                "code": "APV02",
                "displayValue": "One DV",
                "id": "0v6T1000000000uIAA",
                "name": "One",
```

```
            "sequence": "2",
            "value": "One 1",
            "status": "Active"
          },
          {
            "abbreviation": "Red Abbreviation",
            "code": "APV01",
            "displayValue": "Red Display Value",
            "id": "0v6T1000000001OIAQ",
            "name": "Red",
            "sequence": "1",
            "value": "red12",
            "status": "Active"
          }
        ]
      },
      "sequence": 1,
      "status": "Active",
      "valueDescription": "AD Picklist VD"
    }
  ],
  "availabilityDate": "2023-07-12T19:00:00.000Z",
  "categories": [],
  "childProducts": [],
  "configureDuringSale": "Allowed",
  "description": "Bundle Product Description",
  "discontinuedDate": "2023-07-27T19:00:00.000Z",
  "displayUrl": "www.google.com",
  "endOfLifeDate": "2023-07-31T19:00:00.000Z",
  "id": "01tT1000000F0afIAC",
  "isActive": true,
  "isAssetizable": true,
  "isSoldOnlyWithOtherProds": true,
  "name": "Bundle Product",
  "nodeType": "bundleProduct",
  "productClassification": {
    "id": "11BT10000004C9SMAU",
    "name" : "class",
    "code" : "code",
    "status" : "Active"
  },
  "productCode": "P001",
  "productComponentGroups": [
    {
      "code": "PCG002",
      "components": [
        {
          "attributeCategory": [],
          "attributes": [],
          "categories": [],
          "childProducts": [],
          "nodeType": "productClass",
          "productClassification": {
            "id": "11BT10000004C9SMAU",
```

```
            "name" : "class",
            "code" : "code",
            "status" : "Active"
          },
          "productComponentGroups": [],
          "productRelatedComponent": {
            "childSellingModelId": "0jPT10000004CAfMAM",
            "doesBundlePriceIncludeChild": true,
            "id": "0dST100000000rlMAA",
            "isComponentRequired": false,
            "isDefaultComponent": false,
            "isQuantityEditable": true,
            "maxQuantity": 5,
            "minQuantity": 1,
            "parentProductId": "01tT1000000F0afIAC",
            "productClassificationId": "11BT10000004C9SMAU",
            "productRelationshipTypeId": "0yoT10000004CBEIA2",
            "quantity": 1,
            "quantityScaleMethod": "Proportional",
            "sequence": 2,
            "isExcluded": false
          },
          "productSellingModelOptions": []
        }
      ],
      "description": "PCG002 desc",
      "id": "0y7T10000004C9IIAU",
      "maxBundleComponents": 5,
      "minBundleComponents": 1,
      "name": "PCG002",
      "parentProductId": "01tT1000000F0afIAC",
      "sequence": 2,
      "isExcluded": false
    },
    {
      "code": "PCG001",
      "components": [
        {
          "attributeCategory": [],
          "attributes": [
            {
              "attributeNameOverride": "AD Picklist",
              "code": "AD001",
              "dataType": "Picklist",
              "defaultValue": "Red",
              "description": "AD Picklist Description",
              "helpText": "AD Picklist DHT",
              "id": "0tjT1000000002WIAQ",
              "isHidden": false,
              "isPriceImpacting": false,
              "isReadOnly": false,
              "isRequired": true,
              "label": "AD Picklist Label",
              "name": "AD Picklist",
```

```
      "picklist": {
        "dataType": "Text",
        "description": "APV Description",
        "id": "0v5T10000000001IAA",
        "name": "Color",
        "values": [
          {
            "abbreviation": "Blue Abb",
            "code": "APV03",
            "displayValue": "Blue DV",
            "id": "0v6T10000000006IAA",
            "name": "Blue",
            "sequence": "3",
            "value": "Blue b",
            "status": "Active"
          },
          {
            "abbreviation": "Red Abb",
            "code": "APV04",
            "displayValue": "Red",
            "id": "0v6T10000000001IAA",
            "name": "Red",
            "sequence": "4",
            "value": "Red",
            "status": "Active"
          },
          {
            "abbreviation": "One Abb",
            "code": "APV02",
            "displayValue": "One DV",
            "id": "0v6T1000000000uIAA",
            "name": "One",
            "sequence": "2",
            "value": "One 1",
            "status": "Active"
          },
          {
            "abbreviation": "Red Abbreviation",
            "code": "APV01",
            "displayValue": "Red Display Value",
            "id": "0v6T1000000001OIAQ",
            "name": "Red",
            "sequence": "1",
            "value": "red12",
            "status": "Active"
          }
        ]
      },
      "sequence": 1,
      "status": "Active",
      "valueDescription": "AD Picklist VD"
    },
    {
      "attributeNameOverride": "AD Text",
```

```
    "code": "AD02",
    "dataType": "Text",
    "displayType": "Text",
    "MinimumCharacterCount": "1",
    "MaximumCharacterCount": "20",
    "defaultValue": "AD Text DV",
    "description": "AD Text Desc",
    "helpText": "AD Text DHT",
    "id": "0tjT1000000002bIAA",
    "isHidden": false,
    "isPriceImpacting": false,
    "isReadOnly": false,
    "isRequired": true,
    "label": "AD Text Label",
    "name": "AD Text",
    "status": "Active",
    "valueDescription": "AD Text VD"
  }
],
"availabilityDate": "2023-07-17T19:00:00.000Z",
"categories": [],
"childProducts": [],
"configureDuringSale": "Allowed",
"description": "P003 desc",
"discontinuedDate": "2023-07-19T19:00:00.000Z",
"displayUrl": "www.google.com",
"endOfLifeDate": "2023-07-28T19:00:00.000Z",
"id": "01tT1000000F0YyIAK",
"isActive": false,
"isAssetizable": true,
"isSoldOnlyWithOtherProds": false,
"name": "Child1 - Bundle with PCG",
"nodeType": "bundleProduct",
"productClassification": {
  "id": "11BT10000004C9SMAU",
  "name" : "class",
  "code" : "code",
  "status" : "Active"
},
"productCode": "P003",
"productComponentGroups": [
  {
    "code": "PCG2",
    "components": [
      {
        "attributeCategory": [],
        "attributes": [],
        "categories": [],
        "childProducts": [],
        "id": "01tT1000000F0Z8IAK",
        "isActive": false,
        "isAssetizable": true,
        "isSoldOnlyWithOtherProds": false,
        "name": "Super Child2 - Bundle with PCG",
```

```
    "nodeType": "bundleProduct",
    "productComponentGroups": [],
    "productRelatedComponent": {
      "childProductId": "01tT1000000F0Z8IAK",
      "doesBundlePriceIncludeChild": true,
      "id": "0dST100000000rWMAQ",
      "isComponentRequired": false,
      "isDefaultComponent": false,
      "isQuantityEditable": false,
      "parentProductId": "01tT1000000F0YyIAK",
      "productRelationshipTypeId": "0yoT1000000002WIAQ",
      "quantity": 1,
      "quantityScaleMethod": "Proportional",
      "sequence": 2,
      "isExcluded": false
    },
    "productSellingModelOptions": [],
    "productSpecificationType": {
      "name": "NonCommercialSpecType",
      "productSpecificationRecordType": null
    }
  },
  {
    "attributeCategory": [],
    "attributes": [],
    "availabilityDate": "2023-07-15T19:00:00.000Z",
    "categories": [],
    "childProducts": [],
    "configureDuringSale": "Allowed",
    "discontinuedDate": "2023-07-16T19:00:00.000Z",
    "displayUrl": "Test",
    "endOfLifeDate": "2023-07-17T19:00:00.000Z",
    "id": "01tT1000000F0YzIAK",
    "isActive": false,
    "isAssetizable": true,
    "isSoldOnlyWithOtherProds": false,
    "name": "SuperChild1 - Bundle with PCG",
    "nodeType": "bundleProduct",
    "productCode": "Test",
    "productComponentGroups": [],
    "productRelatedComponent": {
      "childProductId": "01tT1000000F0YzIAK",
      "doesBundlePriceIncludeChild": true,
      "id": "0dST100000000rXMAQ",
      "isComponentRequired": false,
      "isDefaultComponent": false,
      "isQuantityEditable": false,
      "parentProductId": "01tT1000000F0YyIAK",
      "productRelationshipTypeId": "0yoT1000000002WIAQ",
      "quantity": 1,
      "quantityScaleMethod": "Proportional",
      "sequence": 1,
      "isExcluded": false
    },
```

```
            "productSellingModelOptions": [],
            "productSpecificationType": {
              "name": "NonCommercialSpecType",
              "productSpecificationRecordType": null
            }
          },
          {
            "attributeCategory": [],
            "attributes": [],
            "categories": [],
            "childProducts": [],
            "configureDuringSale": "Allowed",
            "id": "01tT1000000F0apIAC",
            "isActive": false,
            "isAssetizable": true,
            "isSoldOnlyWithOtherProds": false,
            "name": "Bundle2",
            "nodeType": "bundleProduct",
            "productCode": "PC003",
            "productComponentGroups": [],
            "productRelatedComponent": {
              "childProductId": "01tT1000000F0apIAC",
              "doesBundlePriceIncludeChild": true,
              "id": "0dST100000000rqMAA",
              "isComponentRequired": false,
              "isDefaultComponent": false,
              "isQuantityEditable": false,
              "parentProductId": "01tT1000000F0YyIAK",
              "productRelationshipTypeId": "0yoT1000000002WIAQ",
              "quantity": 1,
              "quantityScaleMethod": "Proportional",
              "isExcluded": false
            },
            "productSellingModelOptions": [],
            "productSpecificationType": {
              "name": "NonCommercialSpecType",
              "productSpecificationRecordType": null
            }
          }
        ],
        "description": "Group for components at level 2",
        "id": "0y7T10000004C98IAE",
        "maxBundleComponents": 5,
        "minBundleComponents": 1,
        "name": "PCG2",
        "parentProductId": "01tT1000000F0YyIAK",
        "isExcluded": false
      }
    ],
    "productRelatedComponent": {
      "childProductId": "01tT1000000F0YyIAK",
      "childSellingModelId": "0jPT10000004CAfMAM",
      "doesBundlePriceIncludeChild": true,
      "id": "0dST100000000rgMAA",
```

```
            "isComponentRequired": false,
            "isDefaultComponent": false,
            "isQuantityEditable": true,
            "maxQuantity": 3,
            "minQuantity": 1,
            "parentProductId": "01tT1000000F0afIAC",
            "parentSellingModelId": "0jPT10000004CAfMAM",
            "productRelationshipTypeId": "0yoT1000000002WIAQ",
            "quantity": 1,
            "quantityScaleMethod": "Proportional",
            "sequence": 1,
            "isExcluded": false
          },
          "productSellingModelOptions": [
            {
              "id": "0iOT10000004CMrMAM",
              "productId": "01tT1000000F0YyIAK",
              "isDefault": false,
              "productSellingModel": {
                "id": "0jPT10000004CAfMAM",
                "name": "OneTimePSM",
                "sellingModelType": "OneTime",
                "status": "Active"
              }
            }
          ],
          "productSpecificationType": {
            "name": "NonCommercialSpecType",
            "productSpecificationRecordType": null
          }
        }
      ],
      "description": "PCG001 Description",
      "id": "0y7T10000004C9DIAU",
      "maxBundleComponents": 5,
      "minBundleComponents": 1,
      "name": "PCG001",
      "parentProductId": "01tT1000000F0afIAC",
      "sequence": 1,
      "isExcluded": false
    }
  ],
  "productSellingModelOptions": [
    {
      "id": "0iOT10000004CMmMAM",
      "productId": "01tT1000000F0afIAC",
      "productSellingModel": {
        "id": "0jPT10000004CAfMAM",
        "name": "OneTimePSM",
        "sellingModelType": "OneTime",
        "status": "Active"
      }
    }
  ],
```

```
      "productSpecificationType": {
        "name": "NonCommercialSpecType",
        "productSpecificationRecordType": null
      }
    },
    {
      "attributeCategory": [
        {
          "attributes": [
            {
              "attributeNameOverride": "AD Text",
              "code": "AD02",
              "dataType": "Text",
              "displayType": "Text",
              "MinimumCharacterCount": "1",
              "MaximumCharacterCount": "20",
              "defaultValue": "AD Text DV",
              "description": "AD Text Desc",
              "helpText": "AD Text DHT",
              "id": "0tjT1000000002bIAA",
              "isHidden": false,
              "isPriceImpacting": true,
              "isReadOnly": true,
              "isRequired": true,
              "label": "AD Text Label",
              "name": "AD Text",
              "sequence": 1,
              "status": "Active",
              "valueDescription": "AD Text VD"
            }
          ],
          "code": "AC001",
          "id": "0v3T1000000000BIAQ",
          "name": "build and make"
        }
      ],
      "attributes": [
        {
          "attributeNameOverride": "AD Picklist",
          "code": "AD001",
          "dataType": "Picklist",
          "defaultValue": "Red",
          "description": "AD Picklist Description",
          "helpText": "AD Picklist DHT",
          "id": "0tjT1000000002WIAQ",
          "isHidden": false,
          "isPriceImpacting": false,
          "isReadOnly": false,
          "isRequired": true,
          "label": "AD Picklist Label",
          "name": "AD Picklist",
          "picklist": {
            "dataType": "Text",
            "description": "APV Description",
```

```
          "id": "0v5T10000000001IAA",
          "name": "Color",
          "values": [
            {
              "abbreviation": "Blue Abb",
              "code": "APV03",
              "displayValue": "Blue DV",
              "id": "0v6T10000000006IAA",
              "name": "Blue",
              "sequence": "3",
              "value": "Blue b",
              "status": "Active"
            },
            {
              "abbreviation": "Red Abb",
              "code": "APV04",
              "displayValue": "Red",
              "id": "0v6T10000000001IAA",
              "name": "Red",
              "sequence": "4",
              "value": "Red",
              "status": "Active"
            },
            {
              "abbreviation": "One Abb",
              "code": "APV02",
              "displayValue": "One DV",
              "id": "0v6T1000000000uIAA",
              "name": "One",
              "sequence": "2",
              "value": "One 1",
              "status": "Active"
            },
            {
              "abbreviation": "Red Abbreviation",
              "code": "APV01",
              "displayValue": "Red Display Value",
              "id": "0v6T1000000001OIAQ",
              "name": "Red",
              "sequence": "1",
              "value": "red12",
              "status": "Active"
            }
          ]
        },
        "sequence": 1,
        "status": "Active",
        "valueDescription": "AD Picklist VD"
      }
    ],
    "availabilityDate": "2023-07-12T19:00:00.000Z",
    "categories": [],
    "childProducts": [],
    "configureDuringSale": "Allowed",
```

```
    "description": "Bundle Product Description",
    "discontinuedDate": "2023-07-27T19:00:00.000Z",
    "displayUrl": "www.google.com",
    "endOfLifeDate": "2023-07-31T19:00:00.000Z",
    "id": "01tT1000000F0afIAC",
    "isActive": true,
    "isAssetizable": true,
    "isSoldOnlyWithOtherProds": true,
    "name": "Bundle Product",
    "nodeType": "bundleProduct",
    "productClassification": {
      "id": "11BT10000004C9SMAU",
      "name" : "class",
      "code" : "code",
      "status" : "Active"
    },
    "productCode": "P001",
    "productComponentGroups": [
      {
        "code": "PCG002",
        "components": [
          {
            "attributeCategory": [],
            "attributes": [],
            "categories": [],
            "childProducts": [],
            "nodeType": "productClass",
            "productClassification": {
              "id": "11BT10000004C9SMAU",
              "name" : "class",
              "code" : "code",
              "status" : "Active"
            },
            "productComponentGroups": [],
            "productRelatedComponent": {
              "childSellingModelId": "0jPT10000004CAfMAM",
              "doesBundlePriceIncludeChild": true,
              "id": "0dST100000000rlMAA",
              "isComponentRequired": false,
              "isDefaultComponent": false,
              "isQuantityEditable": true,
              "maxQuantity": 5,
              "minQuantity": 1,
              "parentProductId": "01tT1000000F0afIAC",
              "productClassificationId": "11BT10000004C9SMAU",
              "productRelationshipTypeId": "0yoT10000004CBEIA2",
              "quantity": 1,
              "quantityScaleMethod": "Proportional",
              "sequence": 2,
              "isExcluded": false
            },
            "productSellingModelOptions": []
          }
        ],
```

```
      "description": "PCG002 desc",
      "id": "0y7T10000004C9IIAU",
      "maxBundleComponents": 5,
      "minBundleComponents": 1,
      "name": "PCG002",
      "parentProductId": "01tT1000000F0afIAC",
      "sequence": 2,
      "isExcluded": false
    },
    {
      "code": "PCG001",
      "components": [
        {
          "attributeCategory": [],
          "attributes": [
            {
              "attributeNameOverride": "AD Picklist",
              "code": "AD001",
              "dataType": "Picklist",
              "defaultValue": "Red",
              "description": "AD Picklist Description",
              "helpText": "AD Picklist DHT",
              "id": "0tjT1000000002WIAQ",
              "isHidden": false,
              "isPriceImpacting": false,
              "isReadOnly": false,
              "isRequired": true,
              "label": "AD Picklist Label",
              "name": "AD Picklist",
              "picklist": {
                "dataType": "Text",
                "description": "APV Description",
                "id": "0v5T10000000001IAA",
                "name": "Color",
                "values": [
                  {
                    "abbreviation": "Blue Abb",
                    "code": "APV03",
                    "displayValue": "Blue DV",
                    "id": "0v6T10000000006IAA",
                    "name": "Blue",
                    "sequence": "3",
                    "value": "Blue b",
                    "status": "Active"
                  },
                  {
                    "abbreviation": "Red Abb",
                    "code": "APV04",
                    "displayValue": "Red",
                    "id": "0v6T10000000001IAA",
                    "name": "Red",
                    "sequence": "4",
                    "value": "Red",
                    "status": "Active"
```

```
            },
            {
              "abbreviation": "One Abb",
              "code": "APV02",
              "displayValue": "One DV",
              "id": "0v6T1000000000uIAA",
              "name": "One",
              "sequence": "2",
              "value": "One 1",
              "status": "Active"
            },
            {
              "abbreviation": "Red Abbreviation",
              "code": "APV01",
              "displayValue": "Red Display Value",
              "id": "0v6T1000000001OIAQ",
              "name": "Red",
              "sequence": "1",
              "value": "red12",
              "status": "Active"
            }
          ]
        },
        "sequence": 1,
        "status": "Active",
        "valueDescription": "AD Picklist VD"
      },
      {
        "attributeNameOverride": "AD Text",
        "code": "AD02",
        "dataType": "Text",
        "displayType": "Text",
        "MinimumCharacterCount": "1",
        "MaximumCharacterCount": "20",
        "defaultValue": "AD Text DV",
        "description": "AD Text Desc",
        "helpText": "AD Text DHT",
        "id": "0tjT1000000002bIAA",
        "isHidden": false,
        "isPriceImpacting": false,
        "isReadOnly": false,
        "isRequired": true,
        "label": "AD Text Label",
        "name": "AD Text",
        "status": "Active",
        "valueDescription": "AD Text VD"
      }
    ],
    "availabilityDate": "2023-07-17T19:00:00.000Z",
    "categories": [],
    "childProducts": [],
    "configureDuringSale": "Allowed",
    "description": "P003 desc",
    "discontinuedDate": "2023-07-19T19:00:00.000Z",
```

```
"displayUrl": "www.google.com",
"endOfLifeDate": "2023-07-28T19:00:00.000Z",
"id": "01tT1000000F0YyIAK",
"isActive": false,
"isAssetizable": true,
"isSoldOnlyWithOtherProds": false,
"name": "Child1 - Bundle with PCG",
"nodeType": "bundleProduct",
"productClassification": {
  "id": "11BT10000004C9SMAU",
  "name" : "class",
  "code" : "code",
  "status" : "Active"
},
"productCode": "P003",
"productComponentGroups": [
  {
    "code": "PCG2",
    "components": [
      {
        "attributeCategory": [],
        "attributes": [],
        "categories": [],
        "childProducts": [],
        "id": "01tT1000000F0Z8IAK",
        "isActive": false,
        "isAssetizable": true,
        "isSoldOnlyWithOtherProds": false,
        "name": "Super Child2 - Bundle with PCG",
        "nodeType": "bundleProduct",
        "productComponentGroups": [],
        "productRelatedComponent": {
          "childProductId": "01tT1000000F0Z8IAK",
          "doesBundlePriceIncludeChild": true,
          "id": "0dST100000000rWMAQ",
          "isComponentRequired": false,
          "isDefaultComponent": false,
          "isQuantityEditable": false,
          "parentProductId": "01tT1000000F0YyIAK",
          "productRelationshipTypeId": "0yoT1000000002WIAQ",
          "quantity": 1,
          "quantityScaleMethod": "Proportional",
          "sequence": 2,
          "isExcluded": false
        },
        "productSellingModelOptions": [],
        "productSpecificationType": {
          "name": "NonCommercialSpecType",
          "productSpecificationRecordType": null
        }
      },
      {
        "attributeCategory": [],
        "attributes": [],
```

```
    "availabilityDate": "2023-07-15T19:00:00.000Z",
    "categories": [],
    "childProducts": [],
    "configureDuringSale": "Allowed",
    "discontinuedDate": "2023-07-16T19:00:00.000Z",
    "displayUrl": "Test",
    "endOfLifeDate": "2023-07-17T19:00:00.000Z",
    "id": "01tT1000000F0YzIAK",
    "isActive": false,
    "isAssetizable": true,
    "isSoldOnlyWithOtherProds": false,
    "name": "SuperChild1 - Bundle with PCG",
    "nodeType": "bundleProduct",
    "productCode": "Test",
    "productComponentGroups": [],
    "productRelatedComponent": {
      "childProductId": "01tT1000000F0YzIAK",
      "doesBundlePriceIncludeChild": true,
      "id": "0dST100000000rXMAQ",
      "isComponentRequired": false,
      "isDefaultComponent": false,
      "isQuantityEditable": false,
      "parentProductId": "01tT1000000F0YyIAK",
      "productRelationshipTypeId": "0yoT1000000002WIAQ",
      "quantity": 1,
      "quantityScaleMethod": "Proportional",
      "sequence": 1,
      "isExcluded": false
    },
    "productSellingModelOptions": [],
    "productSpecificationType": {
      "name": "NonCommercialSpecType",
      "productSpecificationRecordType": null
    }
  },
  {
    "attributeCategory": [],
    "attributes": [],
    "categories": [],
    "childProducts": [],
    "configureDuringSale": "Allowed",
    "id": "01tT1000000F0apIAC",
    "isActive": false,
    "isAssetizable": true,
    "isSoldOnlyWithOtherProds": false,
    "name": "Bundle2",
    "nodeType": "bundleProduct",
    "productCode": "PC003",
    "productComponentGroups": [],
    "productRelatedComponent": {
      "childProductId": "01tT1000000F0apIAC",
      "doesBundlePriceIncludeChild": true,
      "id": "0dST100000000rqMAA",
      "isComponentRequired": false,
```

```
                "isDefaultComponent": false,
                "isQuantityEditable": false,
                "parentProductId": "01tT1000000F0YyIAK",
                "productRelationshipTypeId": "0yoT1000000002WIAQ",
                "quantity": 1,
                "quantityScaleMethod": "Proportional",
                "isExcluded": false
              },
              "productSellingModelOptions": [],
              "productSpecificationType": {
                "name": "NonCommercialSpecType",
                "productSpecificationRecordType": null
              }
            }
          ],
          "description": "Group for components at level 2",
          "id": "0y7T10000004C98IAE",
          "maxBundleComponents": 5,
          "minBundleComponents": 1,
          "name": "PCG2",
          "parentProductId": "01tT1000000F0YyIAK",
          "isExcluded": false
        }
      ],
      "productRelatedComponent": {
        "childProductId": "01tT1000000F0YyIAK",
        "childSellingModelId": "0jPT10000004CAfMAM",
        "doesBundlePriceIncludeChild": true,
        "id": "0dST100000000rgMAA",
        "isComponentRequired": false,
        "isDefaultComponent": false,
        "isQuantityEditable": true,
        "maxQuantity": 3,
        "minQuantity": 1,
        "parentProductId": "01tT1000000F0afIAC",
        "parentSellingModelId": "0jPT10000004CAfMAM",
        "productRelationshipTypeId": "0yoT1000000002WIAQ",
        "quantity": 1,
        "quantityScaleMethod": "Proportional",
        "sequence": 1,
        "isExcluded": false
      },
      "productSellingModelOptions": [
        {
          "id": "0iOT10000004CMrMAM",
          "productId": "01tT1000000F0YyIAK",
          "isDefault": false,
          "productSellingModel": {
            "id": "0jPT10000004CAfMAM",
            "name": "OneTimePSM",
            "sellingModelType": "OneTime",
            "status": "Active"
          }
        }
```

```
              ],
              "productSpecificationType": {
                "name": "NonCommercialSpecType",
                "productSpecificationRecordType": null
              }
            }
          ],
          "description": "PCG001 Description",
          "id": "0y7T10000004C9DIAU",
          "maxBundleComponents": 5,
          "minBundleComponents": 1,
          "name": "PCG001",
          "parentProductId": "01tT1000000F0afIAC",
          "sequence": 1,
          "isExcluded": false
        }
      ],
      "productSellingModelOptions": [
        {
          "id": "0iOT10000004CMmMAM",
          "productId": "01tT1000000F0afIAC",
          "productSellingModel": {
            "id": "0jPT10000004CAfMAM",
            "name": "OneTimePSM",
            "sellingModelType": "OneTime",
            "status": "Active"
          }
        }
      ],
      "productSpecificationType": {
        "name": "NonCommercialSpecType",
        "productSpecificationRecordType": null
      }
    }
  ],
  "status": {
    "code": "200",
    "correlationId": "fd158d80-d73c-4a1f-a009-9225db804d70",
    "errors": [],
    "message": "Successfully fetched Product records."
  }
}
```


Property Name	Type	Description	Filter Group and Version	Available Version
additionalFields	Map<String, Additional Fields Input>	Key-value pair of additional standard or custom fields with their values.	Small, 61.0	61.0
attributeCategory	Attribute Category[]	List of categorized attributes related to the product.	Small, 60.0	60.0

Property Name	Type	Description	Filter Group and Version	Available Version
`attributes`	Attribute Definition[]	List of uncategorized attributes related to the product.	Small, 60.0	60.0
`availabilityDate`	String	Date when the part is used in the product or is made available for sale.	Small, 60.0	60.0
`catalogs`	Catalog[]	List of the associated catalogs returned with the Product List API (POST) response. The Product By ID API (GET) returns an empty catalog list in the response. Returns the `name` and `id` values only.	Small, 61.0	61.0
`categories`	Category[]	List of the associated categories returned with the Product List API (POST) response. The Product By ID API (GET) returns an empty category list in the response. Returns the `name` and `id` values only.	Small, 60.0	60.0
`childProducts`	Product[]	Hierarchy of the child products.	Small, 60.0	60.0
`configureDuringSale`	String	Determines whether to allow or prevent configuration when a bundle is sold.	Small, 60.0	60.0
`description`	String	Description of the product. If data translation is set up and specified in the org, the translated description is available.	Small, 60.0	60.0
`discontinuedDate`	String	Date from when the part can't be used in the product or sold.	Small, 60.0	60.0
`displayUrl`	String	Display image URL of the product.	Small, 60.0	60.0
`endOfLifeDate`	String	Date after which a product isn't supported, ordered, or maintained.	Small, 60.0	60.0
`id`	String	ID of the product.	Small, 60.0	60.0
`isActive`	Boolean	Indicates if the product is active (`true`) or not (`false`).	Small, 60.0	60.0
`isAssetizable`	Boolean	Indicates if the product instance remains a customer asset after it's purchased (`true`) or not (`false`).	Small, 60.0	60.0
`isSoldOnlyWithOtherProds`	Boolean	Indicates whether the product can't be sold separately (`true`) or not (`false`).	Small, 60.0	60.0

Property Name	Type	Description	Filter Group and Version	Available Version
name	String	Name of the product. If data translation is set up and specified in the org, the translated name is available.	Small, 60.0	60.0
nodeType	String	Type of the node, such as a product or bundled product.	Small, 60.0	60.0
productClassification	Product Classification	Details of the product classification that the product is based on.	Small, 60.0	60.0
productCode	String	Universal product code that's used to track the part that's used in the product.	Small, 60.0	60.0
productComponentGroups	Product Component Group[]	Logical grouping of the component products in a bundle and the group cardinality for ordering the product components.	Small, 60.0	60.0
productRelatedComponent	Product Related Component	Details of the related components of a product.	Small, 60.0	60.0
productSellingModelOptions	Product Selling Model Option[]	Details of the product selling model options.	Small, 60.0	60.0
productSpecificationType	Product Specification Type	Details of the product specification type.	Small, 60.0	60.0
quantityScaleMethod	String	Method to scale the quantity of the child product in relation to the quantity of the parent.	Small, 60.0	60.0

Product Related Component

Output representation of the product-related component.

JSON example

```
"productRelatedComponent": {
      "childProductId": "01tT1000000F0YyIAK",
      "childSellingModelId": "0jPT10000004CAfMAM",
      "doesBundlePriceIncludeChild": true,
      "id": "0dST100000000rgMAA",
      "isComponentRequired": false,
      "isDefaultComponent": false,
      "isExcluded": false,
      "isQuantityEditable": true,
      "maxQuantity": 3,
      "minQuantity": 1,
```

```
        "parentProductId": "01tT1000000F0afIAC",
        "parentSellingModelId": "0jPT10000004CAfMAM",
        "productClassificationId": "11BRO00000000222AA",
        "productRelationshipTypeId": "0yoT1000000002WIAQ",
        "quantity": 1,
        "quantityScaleMethod": "Proportional",
        "sequence": 1
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
childProductId	String	Lookup to the child product in the bundle.	Small, 60.0	60.0
childSellingModelId	String	ID of the child product selling model record.	Small, 60.0	60.0
doesBundlePriceIncludeChild	Boolean	Indicates whether the price of the bundle includes the child product (true) or not (false).	Small, 60.0	60.0
id	String	ID of the record.	Small, 60.0	60.0
isComponentRequired	Boolean	Indicates whether the component is required in the bundle (true) or not (false).	Small, 60.0	60.0
isDefaultComponent	Boolean	Indicates whether to select the component in the bundle group by default (true) or not (false).	Small, 60.0	60.0
isExcluded	Boolean	Indicates whether the component is excluded in the bundle group (true) or not (false).	Small, 60.0	60.0
isQuantityEditable	Boolean	Indicates whether to allow changes to the quantity of the component in the bundle (true) or not (false).	Small, 60.0	60.0
maxQuantity	Double	Maximum quantity of the product in the opportunity, quote, or order line item.	Small, 60.0	60.0
minQuantity	Double	Minimum quantity of the product in the opportunity, quote, or order line item.	Small, 60.0	60.0
parentProductId	String	Lookup to the parent product.	Small, 60.0	60.0
parentSellingModelId	String	ID of the product selling model record.	Small, 60.0	60.0
productClassificationId	String	ID of the product classification record.	Small, 60.0	60.0

Property Name	Type	Description	Filter Group and Version	Available Version
productInstanceReuse	String	Reserved for future use.	Small, 62.0	62.0
productRelationshipTypeId	String	ID of the product relationship type record.	Small, 60.0	60.0
quantity	Double	Quantity of the child products.	Small, 60.0	60.0
quantityScaleMethod	String	Method to scale the quantity of the child product in relation to the quantity of the parent.	Small, 60.0	60.0
sequence	Integer	Order in which the child products are displayed.	Small, 60.0	60.0

Product Selling Model

Output representation of the definition of the product selling model.

JSON example

```
"productSellingModel":
{
  "id": "0jPT10000004CAfMAM",
  "name": "OneTimePSM",
  "pricingTerm": 1,
  "pricingTermUnit": "Months",
  "sellingModelType": "TermDefined",
  "status": "Active"
}
}]
```

Property Name	Type	Description	Filter Group and Version	Available Version
id	String	ID of the record.	Small, 60.0	60.0
name	String	Name of the record.	Small, 60.0	60.0
pricingTerm	Integer	Duration of the selling model.	Small, 60.0	60.0
pricingTermUnit	String	Units of the pricing term.	Small, 60.0	60.0
sellingModelType	String	Different models of selling the product. Valid values are: • OneTime • TermDefined • Evergreen	Small, 60.0	60.0

Property Name	Type	Description	Filter Group and Version	Available Version
status	String	Status of the selling model. For example, whether the selling model is active and can be used in transactions.	Small, 60.0	60.0

Product Selling Model Option

Output representation of the definition of the product selling model option.

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.

JSON example

```
"productSellingModelOptions": [
   {
    "id": "0iOT10000004CMrMAM",
    "isDefault": false,
    "productId": "01tT1000000F0YyIAK",
    "productSellingModel": {
    "id": "0jPT10000004CAfMAM",
    "name": "OneTimePSM",
    "pricingTerm": 1,
    "pricingTermUnit": "Months",
    "sellingModelType": "TermDefined",
    "status": "Active"
  }
}]
```

Property Name	Type	Description	Filter Group and Version	Available Version
id	String	ID of the record.	Small, 60.0	60.0
isDefault	Boolean	Indicates whether this model option is the default product selling model option (true) or not (false).	Small, 60.0	60.0
productId	String	ID of the product.	Small, 60.0	60.0
product SellingModel	Product Selling Model	Master-detail field to the product selling model.	Small, 60.0	60.0

Product Specification Type

Output representation of the product specification type.

JSON example

```
"productSpecificationType":
{
```

```
"name": "NonCommercialSpecType",
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
name	String	Name of the product specification type.	Small, 60.0	60.0

Products

Output representation of the list of retrieved products.

JSON example

```
{
  "correlationId": "9b6bc520-3c82-4d6c-a458-47590370681a",
  "facets":[
    {
      "attributeType": "ProductStandard",
      "displayName": "Product Type",
      "displayRank": 1,
      "displayType": "MultiSelect",
      "nameOrId": "Type",
      "values": [
        {
          "displayName": "Simple",
          "nameOrId": "Simple",
          "productCount": 9
        }
      ]
    },
    {
      "attributeType": "ProductStandard",
      "displayName": "Active",
      "displayRank": 2,
      "displayType": "MultiSelect",
      "nameOrId": "IsActive",
      "values": [
        {
          "displayName": "true",
          "nameOrId": "true",
          "productCount": 47
        }
      ]
    }
  ],
  "count": 3,
  "products": [
    {
      "attributeCategory": [],
      "attributes": [],
      "categories": [],
      "childProducts": [],
     "description": "Keep your organization connected with seamless collaboration across
```

```
 distributed teams. No matter where employees are located, organizations are seeking
stronger employee engagement and customer experiences to enable more productivity and
greater business agility. More effective collaboration helps organizations work smarter.",

      "displayUrl":
"https://dispatch.m.io/wp-content/uploads/2023/01/History-of-Webex.png",
      "id": "01t1Q000008CD2eQAG",
      "isActive": true,
      "isAssetizable": true,
      "isSoldOnlyWithOtherProds": false,
      "name": "SmartBytes Collaboration Suite",
      "nodeType": "bundleProduct",
      "productClassification": {
        "id": "11B1Q0000008OMGUA2"
      },
      "productCode": "P0143",
      "productComponentGroups": [],
      "productSellingModelOptions": [
        {
          "id": "0iO1Q0000008OkeUAE",
          "productId": "01t1Q000008CD2eQAG",
          "productSellingModel": {
            "id": "0jP1Q000000CaVFUA0",
            "name": "One Time",
            "sellingModelType": "OneTime",
            "status": "Active"
          }
        }
      ],
      "productSpecificationType": {
        "name": "None"
      }
    },
    {
      "attributeCategory": [],
      "attributes": [],
      "categories": [],
      "childProducts": [],
      "configureDuringSale": "Allowed",
      "displayUrl":
"https://www.cisco.com/c/dam/en/us/products/collateral/wireless/small-business-300-series-wireless-access-points/datasheet-c78-732143.docx/_jcr_content/renditions/datasheet-c78-732143_1.jpg",

      "id": "01t1Q000008CD36QAG",
      "isActive": true,
      "isAssetizable": true,
      "isSoldOnlyWithOtherProds": false,
      "name": "SmartBytes Repeater",
      "nodeType": "simpleProduct",
      "productClassification": {
        "id": "11B1Q0000008OMMUA2"
      },
      "productCode": "BS_R001",
      "productComponentGroups": [],
      "productSellingModelOptions": [
```

```
        {
          "id": "0iO1Q0000008OlsUAE",
          "productId": "01t1Q000008CD36QAG",
          "productSellingModel": {
            "id": "0jP1Q000000CaVFUA0",
            "name": "One Time",
            "sellingModelType": "OneTime",
            "status": "Active"
          }
        }
      ],
      "productSpecificationType": {
        "name": "COM_Offer"
      }
    },
    {
      "attributeCategory": [],
      "attributes": [],
      "categories": [],
      "childProducts": [],
      "configureDuringSale": "Allowed",
      "description": "A converged data center architecture that integrates computing,
networking and storage resources to increase efficiency and enable centralized management.
 UCS products are designed and configured to work together effectively.",
      "displayUrl":
"https://www.cisco.com/c/dam/en/products/collateral/servers-unified-computing/ucs-series-modular-system/sphere-sales-appliance-with-series-w.docx/jcr_content/renditions/sphere-sales-appliance-with-series-w_3.png",

      "id": "01t1Q000008CD2sQAG",
      "isActive": true,
      "isAssetizable": true,
      "isSoldOnlyWithOtherProds": false,
      "name": "SmartBytes Unified Computing System",
      "nodeType": "bundleProduct",
      "productCode": "P0157",
      "productComponentGroups": [],
      "productSellingModelOptions": [
        {
          "id": "0iO1Q0000008OkzUAE",
          "productId": "01t1Q000008CD2sQAG",
          "productSellingModel": {
            "id": "0jP1Q000000CaVFUA0",
            "name": "One Time",
            "sellingModelType": "OneTime",
            "status": "Active"
          }
        },
        {
          "id": "0iO1Q0000008OlWUAU",
          "productId": "01t1Q000008CD2sQAG",
          "productSellingModel": {
            "id": "0jP1Q000000CaVHUA0",
            "name": "Evergreen",
            "pricingTerm": 1,
            "pricingTermUnit": "Months",
```

```
            "sellingModelType": "Evergreen",
            "status": "Active"
          }
        }
      ],
      "productSpecificationType": {
        "name": "None"
      }
    }
  ],
  "status": {
    "code": "200",
    "errors": [],
    "message": "Successfully fetched the product records."
  }
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
correlationId	String	Unique token to track and associate related events or transactions across different components of the application. If unspecified, a Universally Unique Identifier (UUID) is generated.	Small, 60.0	60.0
count	Integer	Total count of the products matching the request query.	Small, 60.0	60.0
facets	Search Facet	Details of the faceted search. This property is applicable if the **Use Indexed Data For Product Listing and Search** toggle from the Product Discovery Settings page from Setup is enabled.	Small, 63.0	63.0
products	Product[]	List of products matching the request query.	Small, 60.0	60.0
status	Status	Status of the request.	Small, 60.0	60.0

Related Object Records

Output representation of the related records for a specified record ID and related object API name.

JSON example

```
      "relatedObjectRecords": [
        {
          "count": 2,
          "records": [
            {
              "SegmentType": "Yearly",
              "DurationType": "Months",
              "TrialDuration": null,
              "ProductSellingModelId": "0jPxx000000001dEAA",
```

```
            "ProductId": "01txx0000006i44AAA",
            "Id": "1FTxx0000004CDtGAM",
            "Name": "PPRS-000000005"
          },
          {
            "SegmentType": "FreeTrial",
            "DurationType": "Days",
            "TrialDuration": null,
            "ProductSellingModelId": "0jPxx000000001dEAA",
            "ProductId": "01txx0000006i44AAA",
            "Id": "1FTxx0000004CFUGA2",
            "Name": "PPRS-000000006"
          }
        ],
        "relatedObjectAPIName": "ProductRampSegment"
      },
      {
        "count": 2,
        "records": [
          {
            "UsageMetricId": "1BRxx0000004CAeGAM",
            "UsageMetricName": "Test Usage Metric 2",
            "UsageDefinitionProductId": null,
            "Label": "PUG-103",
            "Quantity": 100,
            "Id": "1BXxx0000004CCGGA2"
          },
          {
            "UsageMetricId": "1BRxx0000004CCGGA2",
            "UsageMetricName": "Test Usage Metric 3",
            "UsageDefinitionProductId": "01txx0000006i2eAAA",
            "Label": "PUG-105",
            "Quantity": 500,
            "Id": "1BXxx0000004CFUGA2"
          }
        ],
        "relatedObjectAPIName": "ProductUsageGrant"
      }
    ]
```


Property Name	Type	Description	Filter Group and Version	Available Version
`count`	Integer	Total count of the related records that are returned.	Small, 62.0	62.0
`records`	Map<String, Object>	List of related object records.	Small, 62.0	62.0
`relatedObjectAPIName`	String	API name of the related object to return the records for.	Small, 62.0	62.0

Related Records

Output representation of the list of relatedObject records for a specified record ID.

JSON example

```
"relatedRecords": [
  {
    "recordId": "01txx0000006i44AAA",
    "relatedObjectRecords": [
      {
        "count": 2,
        "records": [
          {
            "SegmentType": "Yearly",
            "DurationType": "Months",
            "TrialDuration": null,
            "ProductSellingModelId": "0jPxx000000001dEAA",
            "ProductId": "01txx0000006i44AAA",
            "Id": "1FTxx0000004CDtGAM",
            "Name": "PPRS-000000005"
          },
          {
            "SegmentType": "FreeTrial",
            "DurationType": "Days",
            "TrialDuration": null,
            "ProductSellingModelId": "0jPxx000000001dEAA",
            "ProductId": "01txx0000006i44AAA",
            "Id": "1FTxx0000004CFUGA2",
            "Name": "PPRS-000000006"
          }
        ],
        "relatedObjectAPIName": "ProductRampSegment"
      },
      {
        "count": 2,
        "records": [
          {
            "UsageMetricId": "1BRxx0000004CAeGAM",
            "UsageMetricName": "Test Usage Metric 2",
            "UsageDefinitionProductId": null,
            "Label": "PUG-103",
            "Quantity": 100,
            "Id": "1BXxx0000004CCGGA2"
          },
          {
            "UsageMetricId": "1BRxx0000004CCGGA2",
            "UsageMetricName": "Test Usage Metric 3",
            "UsageDefinitionProductId": "01txx0000006i2eAAA",
            "Label": "PUG-105",
            "Quantity": 500,
            "Id": "1BXxx0000004CFUGA2"
          }
        ],
        "relatedObjectAPIName": "ProductUsageGrant"
      }
```

```
      ]
    },
    {
      "recordId": "01txx0000006i5gAAA",
      "relatedObjectRecords": [
        {
          "count": 4,
          "records": [
            {
              "SegmentType": "Yearly",
              "DurationType": "Months",
              "TrialDuration": null,
              "ProductSellingModelId": "0jPxx0000000001EAA",
              "ProductId": "01txx0000006i5gAAA",
              "Id": "1FTxx0000004C92GAE",
              "Name": "PPRS-000000001"
            },
            {
              "SegmentType": "Custom",
              "DurationType": "Days",
              "TrialDuration": null,
              "ProductSellingModelId": "0jPxx0000000001EAA",
              "ProductId": "01txx0000006i5gAAA",
              "Id": "1FTxx0000004CAeGAM",
              "Name": "PPRS-000000002"
            },
            {
              "SegmentType": "FreeTrial",
              "DurationType": "Months",
              "TrialDuration": 6,
              "ProductSellingModelId": "0jPxx0000000001EAA",
              "ProductId": "01txx0000006i5gAAA",
              "Id": "1FTxx0000004CCGGA2",
              "Name": "PPRS-000000003"
            },
            {
              "SegmentType": "Custom",
              "DurationType": null,
              "TrialDuration": null,
              "ProductSellingModelId": "0jPxx0000000001EAA",
              "ProductId": "01txx0000006i5gAAA",
              "Id": "1FTxx0000004CDsGAM",
              "Name": "PPRS-000000004"
            }
          ],
          "relatedObjectAPIName": "ProductRampSegment"
        },
        {
          "count": 0,
          "records": [],
          "relatedObjectAPIName": "ProductUsageGrant"
        }
      ]
```

```
    }
  ]
```

Property Name	Type	Description	Filter Group and Version	Available Version
recordId	String	ID of the record.	Small, 62.0	62.0
relatedObjectRecords	Related Object Records[]	List of related object records.	Small, 62.0	62.0

Related Records List

Output representation of the list of related records.

JSON example

```
{
  "correlationId": "f4d6b42a-d9b7-49c9-8fa8-1c7bb6fe99aa",
  "relatedRecords": [
    {
      "recordId": "01txx0000006i44AAA",
      "relatedObjectRecords": [
        {
          "count": 2,
          "records": [
            {
              "SegmentType": "Yearly",
              "DurationType": "Months",
              "TrialDuration": null,
              "ProductSellingModelId": "0jPxx000000001dEAA",
              "ProductId": "01txx0000006i44AAA",
              "Id": "1FTxx0000004CDtGAM",
              "Name": "PPRS-000000005"
            },
            {
              "SegmentType": "FreeTrial",
              "DurationType": "Days",
              "TrialDuration": null,
              "ProductSellingModelId": "0jPxx000000001dEAA",
              "ProductId": "01txx0000006i44AAA",
              "Id": "1FTxx0000004CFUGA2",
              "Name": "PPRS-000000006"
            }
          ],
          "relatedObjectAPIName": "ProductRampSegment"
        },
        {
          "count": 2,
          "records": [
            {
              "UsageMetricId": "1BRxx0000004CAeGAM",
              "UsageMetricName": "Test Usage Metric 2",
              "UsageDefinitionProductId": null,
```

```
            "Label": "PUG-103",
            "Quantity": 100,
            "Id": "1BXxx0000004CCGGA2"
          },
          {
            "UsageMetricId": "1BRxx0000004CCGGA2",
            "UsageMetricName": "Test Usage Metric 3",
            "UsageDefinitionProductId": "01txx0000006i2eAAA",
            "Label": "PUG-105",
            "Quantity": 500,
            "Id": "1BXxx0000004CFUGA2"
          }
        ],
        "relatedObjectAPIName": "ProductUsageGrant"
      }
    ]
  },
  {
    "recordId": "01txx0000006i5gAAA",
    "relatedObjectRecords": [
      {
        "count": 4,
        "records": [
          {
            "SegmentType": "Yearly",
            "DurationType": "Months",
            "TrialDuration": null,
            "ProductSellingModelId": "0jPxx0000000001EAA",
            "ProductId": "01txx0000006i5gAAA",
            "Id": "1FTxx0000004C92GAE",
            "Name": "PPRS-000000001"
          },
          {
            "SegmentType": "Custom",
            "DurationType": "Days",
            "TrialDuration": null,
            "ProductSellingModelId": "0jPxx0000000001EAA",
            "ProductId": "01txx0000006i5gAAA",
            "Id": "1FTxx0000004CAeGAM",
            "Name": "PPRS-000000002"
          },
          {
            "SegmentType": "FreeTrial",
            "DurationType": "Months",
            "TrialDuration": 6,
            "ProductSellingModelId": "0jPxx0000000001EAA",
            "ProductId": "01txx0000006i5gAAA",
            "Id": "1FTxx0000004CCGGA2",
            "Name": "PPRS-000000003"
          },
          {
            "SegmentType": "Custom",
            "DurationType": null,
            "TrialDuration": null,
```

```
              "ProductSellingModelId": "0jPxx0000000001EAA",
              "ProductId": "01txx0000006i5gAAA",
              "Id": "1FTxx0000004CDsGAM",
              "Name": "PPRS-000000004"
            }
          ],
          "relatedObjectAPIName": "ProductRampSegment"
        },
        {
          "count": 0,
          "records": [],
          "relatedObjectAPIName": "ProductUsageGrant"
        }
      ]
    }
  ],
  "status": {
    "code": "200",
    "errors": [
    ],
    "message": ""
  }
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
correlationId	String	Unique token to track and associate related events or transactions across different components of the application. If unspecified, a Universally Unique Identifier (UUID) is generated.	Small, 62.0	62.0
relatedRecords	Related Records[]	List of related records.	Small, 62.0	62.0
status	Status[]	Status of the API request.	Small, 62.0	62.0

Search Facet

Output representation of the details of the faceted search.

JSON example

```
"facets":[
  {
    "attributeType": "ProductStandard",
    "displayName": "Product Type",
    "displayRank": 1,
    "displayType": "MultiSelect",
    "nameOrId": "Type",
    "values": [
      {
```

```
          "displayName": "Simple",
          "nameOrId": "Simple",
          "productCount": 9
        }
      ]
    },
    {
      "attributeType": "ProductStandard",
      "displayName": "Active",
      "displayRank": 2,
      "displayType": "MultiSelect",
      "nameOrId": "IsActive",
      "values": [
        {
          "displayName": "true",
          "nameOrId": "true",
          "productCount": 47
        }
      ]
    }
  ]
```

Property Name	Type	Description	Filter Group and Version	Available Version
attributeType	String	Search attribute type of the facet.	Small, 63.0	63.0
displayName	String	Display name of the facet.	Small, 63.0	63.0
displayRank	Integer	Display rank of the facet.	Small, 63.0	63.0
displayType	String	Display type of the face.	Small, 63.0	63.0
nameOrId	String	Facet name or ID. Reserved for internal use.	Small, 63.0	63.0
values	Facet Value[]	Values of the facet found in the search result. Sorted by display name in alphabetical order.	Small, 63.0	63.0

Setting

Output representation of the setting that's used in indexing.

JSON example

```
"setting": {
  "defaultLanguage": "en_US",
  "id": "1JySG0000000GUb0AM",
  "supportedLanguages": ["en_US","ja","es","nl_NL"]
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`defaultLanguage`	String	Default language for the API.	Small, 63.0	63.0
`id`	String	ID of the setting.	Small, 63.0	63.0
`supportedLanguages`	String[]	List of supported language locales for indexing.	Small, 63.0	63.0

Setting Metadata

Output representation of the metadata associated with a setting.

JSON example

```
"metadata": {
  "activeLanguages": [
    "en_US"
  ]
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`activeLanguages`	String[]	List of active languages in an org.	Small, 63.0	63.0

Snapshot

Output representation of the list of active snapshots.

JSON example

```
"snapshots": [
  {
    "activationDate": "2024-11-06T06:58:02.000Z",
    "activationStatus": "ACTIVE",
    "activationType": "IMMEDIATE",
    "id": "1Avxx0000004C92CAE",
    "snapshotIndexes": [
      {
        "createdDate": "2024-11-06T06:56:49.000Z",
        "id": "1D6xx0000004C92CAE",
        "indexBuildType": "FULL",
        "indexInfos": [
          {
            "buildType": "FULL",
            "id": "0axxx00000000T3AAI",
            "isIncrementable": true,
            "usageType": "LIVE"
          }
```

```
          ],
          "indexLogs": [
            {
              "catalogSnapshotTime": "2024-11-06T16:14:30.000Z",
              "completionTime": "2024-11-06T16:16:02.000Z",
              "createdById": "005xx000001X7x7AAC",
              "indexBuildStatus": "COMPLETED",
              "indexBuildType": "FULL",
              "indexId": "0axxx00000000T3AAI",
              "numberOfChanges": 7
            },
            {
              "catalogSnapshotTime": "2024-11-06T15:03:32.000Z",
              "completionTime": "2024-11-06T15:05:02.000Z",
              "createdById": "005xx000001X7x7AAC",
              "indexBuildStatus": "COMPLETED_WITH_ERRORS",
              "indexBuildType": "INCREMENTAL",
              "indexId": "0axxx00000000RRAAY",
              "message": "Warning: Product errors found.",
              "numberOfChanges": 3
            },
            {
              "catalogSnapshotTime": "2024-11-06T12:35:34.000Z",
              "completionTime": "2024-11-06T12:35:34.000Z",
              "createdById": "005xx000001X7x7AAC",
              "indexBuildStatus": "COMPLETED",
              "indexBuildType": "INCREMENTAL",
              "indexId": "0axxx00000000RRAAY",
              "message": "There are no changes for the partial update.",
              "numberOfChanges": 0
            },
            {
              "catalogSnapshotTime": "2024-11-06T12:07:32.000Z",
              "completionTime": "2024-11-06T12:09:02.000Z",
              "createdById": "005xx000001X7x7AAC",
              "indexBuildStatus": "COMPLETED_WITH_ERRORS",
              "indexBuildType": "FULL",
              "message": "Warning: Product errors found.",
              "numberOfChanges": 1
            }
          ],
          "indexType": "PRODUCT",
          "lastBuildStatus": "IN_PROGRESS",
          "venueId": "1D6xx0000004C92CAE"
        }
      ]
    }
  ]
```


Property Name	Type	Description	Filter Group and Version	Available Version
`activationDate`	String	Activation date of the snapshot.	Small, 62.0	62.0

Property Name	Type	Description	Filter Group and Version	Available Version
activationStatus	String	Activation status of the snapshot. Valid values are: • NONE • ACTIVE—Snapshot is in active status. • EXPIRED—Snapshot is in expired status.	Small, 62.0	62.0
activationType	String	Activation type of the snapshot. Valid values are: • IMMEDIATE—Snapshot is activated immediately after a successful build.	Small, 62.0	62.0
id	String	ID of the snapshot.	Small, 61.0	61.0
snapshotIndexes	Snapshot Index[]	List of indexes created in the snapshot.	Small, 62.0	62.0

Snapshot Collection

Output representation of the retrieved snapshot collection.

JSON example

```
{
  "errors": [],
  "snapshots": [
    {
      "activationDate": "2024-11-06T06:58:02.000Z",
      "activationStatus": "ACTIVE",
      "activationType": "IMMEDIATE",
      "id": "1Avxx0000004C92CAE",
      "snapshotIndexes": [
        {
          "createdDate": "2024-11-06T06:56:49.000Z",
          "id": "1D6xx0000004C92CAE",
          "indexBuildType": "FULL",
          "indexInfos": [
            {
              "buildType": "FULL",
              "id": "0axxx00000000T3AAI",
              "isIncrementable": true,
              "usageType": "LIVE"
            }
          ],
          "indexLogs": [
            {
              "catalogSnapshotTime": "2024-11-06T16:14:30.000Z",
              "completionTime": "2024-11-06T16:16:02.000Z",
              "createdById": "005xx000001X7x7AAC",
```

```
              "indexBuildStatus": "COMPLETED",
              "indexBuildType": "FULL",
              "indexId": "0axxx00000000T3AAI",
              "numberOfChanges": 7
            },
            {
              "catalogSnapshotTime": "2024-11-06T15:03:32.000Z",
              "completionTime": "2024-11-06T15:05:02.000Z",
              "createdById": "005xx000001X7x7AAC",
              "indexBuildStatus": "COMPLETED_WITH_ERRORS",
              "indexBuildType": "INCREMENTAL",
              "indexId": "0axxx00000000RRAAY",
              "message": "Warning: Product errors found.",
              "numberOfChanges": 3
            },
            {
              "catalogSnapshotTime": "2024-11-06T12:35:34.000Z",
              "completionTime": "2024-11-06T12:35:34.000Z",
              "createdById": "005xx000001X7x7AAC",
              "indexBuildStatus": "COMPLETED",
              "indexBuildType": "INCREMENTAL",
              "indexId": "0axxx00000000RRAAY",
              "message": "There are no changes for the partial update.",
              "numberOfChanges": 0
            },
            {
              "catalogSnapshotTime": "2024-11-06T12:07:32.000Z",
              "completionTime": "2024-11-06T12:09:02.000Z",
              "createdById": "005xx000001X7x7AAC",
              "indexBuildStatus": "COMPLETED_WITH_ERRORS",
              "indexBuildType": "FULL",
              "message": "Warning: Product errors found.",
              "numberOfChanges": 1
            }
          ],
          "indexType": "PRODUCT",
          "lastBuildStatus": "IN_PROGRESS",
          "venueId": "1D6xx0000004C92CAE"
        }
      ]
    }
  ],
  "statusCode": "200"
}
```


Property Name	Type	Description	Filter Group and Version	Available Version
errors	Error Output[]	List of errors, if any.	Small, 62.0	62.0
snapshots	Snapshot[]	List of active snapshots.	Small, 62.0	62.0
statusCode	String	Code indicating the status of the request.	Small, 62.0	62.0

Snapshot Deployment

Output representation of the snapshot deployment.

JSON example

This example shows a sample response to the request to build a new snapshot with immediate activation.

```
{
  "errors": [],
  "snapshot": {
    "activationStatus": "NONE",
    "activationType": "IMMEDIATE",
    "id": "1Avxx0000004CFU",
    "snapshotIndexes": [
      {
        "createdDate": "2024-07-24T21:10:48.000Z",
        "id": "1D6xx0000004CFU",
        "indexBuildType": "FULL",
        "indexType": "PRODUCT",
        "lastBuildStatus": "IN_PROGRESS"
      }
    ]
  },
  "statusCode": "200"
}
```

This example shows a sample response of the request to rebuild a snapshot in the `active` status.

```
{
  "errors": [],
  "snapshot": {
    "activationStatus": "NONE",
    "activationType": "IMMEDIATE",
    "id": "1Avxx0000004CH6",
    "snapshotIndexes": [
      {
        "createdDate": "2024-07-24T21:13:05.000Z",
        "id": "1D6xx0000004CH6",
        "indexBuildType": "FULL",
        "indexType": "PRODUCT",
        "lastBuildStatus": "IN_PROGRESS"
      }
    ]
  },
  "statusCode": "200"
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`errors`	Error Output[]	List of errors, if any.	Small, 62.0	62.0
`snapshot`	Snapshot[]	Run-time catalog snapshot associated with the created index.	Small, 62.0	62.0
`statusCode`	String	Code indicating the status of the request.	Small, 62.0	62.0

Snapshot Index

Output representation of the snapshot index of a run-time catalog.

JSON example

```
"snapshotIndexes": [
  {
    "createdDate": "2024-11-06T06:56:49.000Z",
    "id": "1D6xx0000004C92CAE",
    "indexBuildType": "FULL",
    "indexInfos": [
      {
        "buildType": "FULL",
        "id": "0axxx00000000T3AAI",
        "isIncrementable": true,
        "usageType": "LIVE"
      }
    ],
    "indexLogs": [
      {
        "catalogSnapshotTime": "2024-11-06T16:14:30.000Z",
        "completionTime": "2024-11-06T16:16:02.000Z",
        "createdById": "005xx000001X7x7AAC",
        "indexBuildStatus": "COMPLETED",
        "indexBuildType": "FULL",
        "indexId": "0axxx00000000T3AAI",
        "numberOfChanges": 7
      },
      {
        "catalogSnapshotTime": "2024-11-06T15:03:32.000Z",
        "completionTime": "2024-11-06T15:05:02.000Z",
        "createdById": "005xx000001X7x7AAC",
        "indexBuildStatus": "COMPLETED_WITH_ERRORS",
        "indexBuildType": "INCREMENTAL",
        "indexId": "0axxx00000000RRAAY",
        "message": "Warning: Product errors found.",
        "numberOfChanges": 3
      },
      {
        "catalogSnapshotTime": "2024-11-06T12:35:34.000Z",
        "completionTime": "2024-11-06T12:35:34.000Z",
        "createdById": "005xx000001X7x7AAC",
        "indexBuildStatus": "COMPLETED",
        "indexBuildType": "INCREMENTAL",
        "indexId": "0axxx00000000RRAAY",
        "message": "There are no changes for the partial update.",
        "numberOfChanges": 0
      },
      {
        "catalogSnapshotTime": "2024-11-06T12:07:32.000Z",
        "completionTime": "2024-11-06T12:09:02.000Z",
        "createdById": "005xx000001X7x7AAC",
        "indexBuildStatus": "COMPLETED_WITH_ERRORS",
        "indexBuildType": "FULL",
        "message": "Warning: Product errors found.",
```

```
            "numberOfChanges": 1
          }
        ],
        "indexType": "PRODUCT",
        "lastBuildStatus": "IN_PROGRESS",
        "venueId": "1D6xx0000004C92CAE"
      }
    ]
```

Property Name	Type	Description	Filter Group and Version	Available Version
completedDate	String	Completed date of the snapshot index.	Small, 62.0	62.0
createdDate	String	Created date of the snapshot index.	Small, 62.0	62.0
id	String	ID of the snapshot index.	Small, 62.0	62.0
indexBuildType	String	Build type of the snapshot index. Valid value is: • FULL—Specifies a full index build. • INCREMENTAL—Specifies an incremental index build. Available in API version 63.0 and later.	Small, 62.0	62.0
indexInfos	Index Info	Index information records associated with the snapshot index.	Small, 63.0	63.0
indexLogs	Index Logs[]	Index logs associated with the snapshot index.	Small, 63.0	63.0
indexType	String	Index type of the snapshot index. Valid value is: • PRODUCT—Snapshot index is of product type.	Small, 62.0	62.0
lastBuildStatus	String	Last build status of the snapshot index. Valid values are: • IN_PROGRESS—Snapshot index build is in progress. • FAILED—Snapshot index build failed. • COMPLETED—Snapshot index build completed successfully.	Small, 62.0	62.0
numberOfRecords	Integer	Number of indexed records.	Small, 62.0	62.0
venueId	String	Venue ID of the snapshot index.	Small, 63.0	63.0

Snapshot Index Error

Output representation of the error details related to a snapshot index.

JSON example

```
{
  "errors": [],
  "indexErrorDetails": {
    "errorFileId": "069xx0000004C92AAE",
    "indexCreatedDate": "2024-10-03T05:24:18.000Z",
    "indexErrorsCount": 1,
    "indexLastUpdatedDate": "2024-10-03T05:27:00.000Z",
    "itemLevelErrorsCount": 1
  },
  "statusCode": "200"
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`errors`	Error Output[]	List of errors encountered during the processing of the API request.	Small, 63.0	63.0
`indexErrorDetails`	Index Error[]	Count and details of errors that occurred during indexing.	Small, 63.0	63.0
`statusCode`	String	Code that indicates the status of the API request.	Small, 63.0	63.0

Snapshot Index Info

Output representation of the details of a snapshot index.

JSON example

```
 "indexInfos": [
            {
              "buildType": "FULL",
              "id": "0axxx00000000T3AAI",
              "isIncrementable": true,
              "usageType": "LIVE"
            }
          ]
```

Property Name	Type	Description	Filter Group and Version	Available Version
`buildType`	String	Build type of the index. Valid values are: • `FULL`—Specifies a full index build. • `INCREMENTAL`—Specifies an incremental index build. Available in API version 63.0 and later.	Small, 63.0	63.0

Property Name	Type	Description	Filter Group and Version	Available Version
`id`	String	ID of the index information record.	Small, 63.0	63.0
`isIncrementable`	Boolean	Indicates whether a partial build is enabled (`true`) or disabled (`false`).	Small, 63.0	63.0
`usageType`	String	Usage type of the index. Valid values are: • `LIVE`—Index usage type is live. • `OUT_OF_USE`—Index usage type is out of use.	Small, 63.0	63.0

Snapshot Index Log

Output representation of a snapshot index log.

JSON example

```
"indexLogs": [
  {
    "catalogSnapshotTime": "2024-11-06T16:14:30.000Z",
    "completionTime": "2024-11-06T16:16:02.000Z",
    "createdById": "005xx000001X7x7AAC",
    "indexBuildStatus": "COMPLETED",
    "indexBuildType": "FULL",
    "indexId": "0axxx00000000T3AAI",
    "numberOfChanges": 7
  },
  {
    "catalogSnapshotTime": "2024-11-06T15:03:32.000Z",
    "completionTime": "2024-11-06T15:05:02.000Z",
    "createdById": "005xx000001X7x7AAC",
    "indexBuildStatus": "COMPLETED_WITH_ERRORS",
    "indexBuildType": "INCREMENTAL",
    "indexId": "0axxx00000000RRAAY",
    "message": "Warning: Product errors found.",
    "numberOfChanges": 3
  },
  {
    "catalogSnapshotTime": "2024-11-06T12:35:34.000Z",
    "completionTime": "2024-11-06T12:35:34.000Z",
    "createdById": "005xx000001X7x7AAC",
    "indexBuildStatus": "COMPLETED",
    "indexBuildType": "INCREMENTAL",
    "indexId": "0axxx00000000RRAAY",
    "message": "There are no changes for the partial update.",
    "numberOfChanges": 0
  },
  {
    "catalogSnapshotTime": "2024-11-06T12:07:32.000Z",
    "completionTime": "2024-11-06T12:09:02.000Z",
    "createdById": "005xx000001X7x7AAC",
```

```
            "indexBuildStatus": "COMPLETED_WITH_ERRORS",
            "indexBuildType": "FULL",
            "message": "Warning: Product errors found.",
            "numberOfChanges": 1
          }
        ]
```

Property Name	Type	Description	Filter Group and Version	Available Version
catalogSnapshotTime	String	Catalog snapshot time of the index.	Small, 63.0	63.0
completionTime	String	Completion time of the index.	Small, 63.0	63.0
createdById	String	ID of the user that initiated the index build.	Small, 63.0	63.0
indexBuildStatus	String	Status of the index build.	Small, 63.0	63.0
indexBuildType	String	Type of the index build. Valid values are: • FULL—Specifies a full index build. • INCREMENTAL—Specifies an incremental index build. Available in API version 63.0 and later.	Small, 63.0	63.0
indexId	String	ID of the index.	Small, 63.0	63.0
message	String	Message for the index status.	Small, 63.0	63.0
numberOfChanges	Integer	Number of new or changed products included in the index.	Small, 63.0	63.0

Status

Output representation of the status of the request.

JSON example

```
"status": {
    "code": "200",
    "errors": [],
    "message": "Successfully fetched the catalog records."
  }
```

Property Name	Type	Description	Filter Group and Version	Available Version
code	String	Code of the error message.	Small, 60.0	60.0
errors	Error[]	Details of the error.	Small, 60.0	60.0

Property Name	Type	Description	Filter Group and Version	Available Version
message	String	Error message.	Small, 60.0	60.0

Unit of Measure Error

Output representation of the details of errors encountered during the processing of the Unit of Measure API request.

JSON example

```
"errorCodeToErrorMap": {
  "UNIT_OF_MEASURE_INFO_INVALID_UOM_IDS": {
    "errorCode": "UOM_INFO_API_003",
    "messageDetail": "Invalid uomId is passed. Please specify a valid uomId.",
    "messageTitle": "Invalid uomId is passed.",
    "recordIds": [
      "sample"
    ],
    "source": "Unit_Of_Measure_Info_Api"
  }
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
errorCode	String	Localized error code.	Small, 63.0	63.0
messageDetail	String	Localized details of the error message.	Small, 63.0	63.0
messageTitle	String	Localized title of the error message.	Small, 63.0	63.0
recordIds	String[]	List of erroneous record IDs.	Small, 63.0	63.0
source	String	Localized source of the error.	Small, 63.0	63.0

Unit of Measure Info

Output representation of the details of a unit of measure record.

JSON example

```
"uomIdToUnitOfMeasureInfo": {
  "0hEU200000003M5MAI": {
    "id": "0hEU200000003M5MAI",
    "name": "Pounds",
    "roundingMethod": "Nearest",
    "scale": 1,
    "unitCode": "Pounds"
  },
  "0hEU200000003KTMAY": {
    "id": "0hEU200000003KTMAY",
    "name": "Grams",
    "roundingMethod": "Down",
    "scale": 5,
```

```
      "unitCode": "Grams"
    }
  }
```

Property Name	Type	Description	Filter Group and Version	Available Version
id	String	ID of the unit of measure record.	Small, 63.0	63.0
name	String	Name of the unit of measure record.	Small, 63.0	63.0
roundingMethod	String	Data rounding method of the unit of measure record.	Small, 63.0	63.0
scale	Integer	Scale of the unit of measure record.	Small, 63.0	63.0
unitCode	String	Code of the unit of measure record.	Small, 63.0	63.0

Unit of Measure Status

Output representation of the status of the Unit of Measure API request.

JSON example

```
"status": {
  "errors": [],
  "httpStatusCode": "200",
  "message": " Successfully fetched UnitOfMeasure Info. "
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
errors	Unit Of Measure Error[]	Errors encountered during the processing of the API request.	Small, 63.0	63.0
httpStatusCode	String	HTTP status code of the API request.	Small, 63.0	63.0
message	String	Localized response message.	Small, 63.0	63.0

Product Catalog Management Metadata API Types

Metadata API enables you to access some types and feature settings that you can customize in the user interface.

ProductSpecificationType

Represents the specification types in your org that define products with unique terminology specific to the industry.

ProductSpecificationRecType

Represents the association of a product specification type with record types defined on the Product object. The product specification record type also determines if the product specification is sold commercially or not.

SEE ALSO:

Metadata API Developer Guide: Understanding Metadata API

ProductSpecificationType

Represents the specification types in your org that define products with unique terminology specific to the industry.

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.

Parent Type

This type extends the Metadata metadata type and inherits its `fullName` field.

File Suffix and Directory Location

ProductSpecificationType components have the suffix `.productSpecificationType` and are stored in the `productSpecificationTypes` folder.

Version

ProductSpecificationType components are available in API version 60.0 and later.

Special Access Rules

Ensure Product Catalog Management is enabled to access this metadata type.

Fields

Field Name	Description
`description`	**Field Type** string **Description** Required. Description of the product specification type.
`masterLabel`	**Field Type** string **Description** Required.

Field Name	Description
	A user-friendly name for the product specification record type, which is defined when the metadata component is created.

Declarative Metadata Sample Definition

The following is an example of a ProductSpecificationType component.

```
<ProductSpecificationType xmlns="http://soap.sforce.com/2006/04/metadata">
    <masterLabel>sample</masterLabel>
    <description>Sample Description</description>
</ProductSpecificationType>
```

The following is an example `package.xml` that references the previous definition.

```
<Package xmlns="http://soap.sforce.com/2006/04/metadata">
    <types>
        <members>*</members>
        <name>ProductSpecificationType</name>
    </types>
    <version>60.0</version>
</Package>
```

Wildcard Support in the Manifest File

This metadata type supports the wildcard character `*` (asterisk) in the `package.xml` manifest file. For information about using the manifest file, see Deploying and Retrieving Metadata with the Zip File.

ProductSpecificationRecType

Represents the association of a product specification type with record types defined on the Product object. The product specification record type also determines if the product specification is sold commercially or not.

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.

Parent Type

This type extends the Metadata metadata type and inherits its `fullName` field.

File Suffix and Directory Location

ProductSpecificationRecType components have the suffix `.productSpecificationRecType` and are stored in the `productSpecificationRecTypes` folder.

Version

ProductSpecificationRecType components are available in API version 60.0 and later.

Special Access Rules

Ensure Product Catalog Management is enabled to access this metadata type.

Fields

Field Name	Description
`isCommercial`	**Field Type** boolean **Description** Required. Indicates whether the product is sold commercially (`true`) or not (`false`). The default value is `true`.
`masterLabel`	**Field Type** string **Description** Required. A user-friendly name for the product specification record type, which is defined when the metadata component is created.
`productSpecificationType`	**Field Type** string **Description** Required. Product specification type that's associated with the record type. This field is unique within your organization.
`recordType`	**Field Type** string **Description** Required. Custom record type of Product2 object.

Declarative Metadata Sample Definition

The following is an example of a ProductSpecificationRecType component.

```
<ProductSpecificationRecType xmlns="http://soap.sforce.com/2006/04/metadata">
    <masterLabel>sample</masterLabel>
   <recordType>Product2.Offer</recordType>
   <productSpecificationType>Placeholder</productSpecificationType>
   <isCommercial>true</isCommercial>
</ProductSpecificationRecType>
```

The following is an example `package.xml` that references the previous definition.

```
<Package xmlns="http://soap.sforce.com/2006/04/metadata">
    <types>
        <members>*</members>
        <name>ProductSpecificationRecType</name>
    </types>
    <types>
        <members>*</members>
        <name>ProductSpecificationType</name>
    </types>
    <types>
        <members>Product2.Offer</members>
        <name>RecordType</name>
    </types>
    <version>60.0</version>
</Package>
```

Wildcard Support in the Manifest File

This metadata type supports the wildcard character `*` (asterisk) in the `package.xml` manifest file. For information about using the manifest file, see Deploying and Retrieving Metadata with the Zip File.

Product Catalog Management Tooling API Objects

Tooling API exposes metadata used in developer tooling that you can access through REST or SOAP. Tooling API's SOQL capabilities for many metadata types allow you to retrieve smaller pieces of metadata.

ProductSpecificationType

Represents the specification types in your org that define products with unique terminology specific to the industry. This object is available in API version 60.0 and later.

ProductSpecificationRecType

Represents the association of a product specification type with record types defined on the Product object. The product specification record type also determines if the product specification is sold commercially or not. This object is available in API version 60.0 and later.

SEE ALSO:

Tooling API Developer Guide: Introducing Tooling API

ProductSpecificationType

Represents the specification types in your org that define products with unique terminology specific to the industry. This object is available in API version 60.0 and later.

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.

Supported SOAP API Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Supported REST API Methods

`DELETE, GET, HEAD, PATCH, POST, Query`

Special Access Rules

Ensure Product Catalog Management is enabled to access this object.

Fields

Field	Details
`Description`	**Type** textarea **Properties** Create, Nillable, Update **Description** Description of the product specification type.
`DeveloperName`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Label is **Record Type Name**.
`Language`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Language of the product specification type instance. Possible values are: • `da`—Danish • `de`—German • `en_US`—English

Field	Details
	• es—Spanish • es_MX—Spanish (Mexico) • fi—Finnish • fr—French • it—Italian • ja—Japanese • ko—Korean • nl_NL—Dutch • no—Norwegian • pt_BR—Portuguese (Brazil) • ru—Russian • sv—Swedish • th—Thai • zh_CN—Chinese (Simplified) • zh_TW—Chinese (Traditional)
ManageableState	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** **Type** ManageableState enumerated list **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** Indicates the manageable state of the specified component that is contained in a package: • beta • deleted • deprecated • deprecatedEditable • installed • installedEditable • released • unmanaged
MasterLabel	**Type** string

Field	Details
	Properties Create, Filter, Group, Sort, Update **Description** Label assigned to the ProductSpecificationType object.
`NamespacePrefix`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the ***`namespacePrefix__componentName`*** notation. The namespace prefix can have one of the following values: • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are not Developer Edition organizations, `NamespacePrefix` is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.

ProductSpecificationRecType

Represents the association of a product specification type with record types defined on the Product object. The product specification record type also determines if the product specification is sold commercially or not. This object is available in API version 60.0 and later.

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.

Supported SOAP API Calls

`create(), delete(), describeSObjects(), query(), retrieve(), update(), upsert()`

Supported REST API Methods

`DELETE, GET, HEAD, PATCH, POST, Query`

Special Access Rules

Ensure Product Catalog Management is enabled to access this object.

Fields

Field	Details
`DeveloperName`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Label is **Record Type Name**.
`IsCommercial`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the product is sold commercially (`true`) or not (`false`). The default value is `true`.
`Language`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Language of the product specification record type instance. Possible values are: • `da`—Danish • `de`—German • `en_US`—English • `es`—Spanish • `es_MX`—Spanish (Mexico) • `fi`—Finnish • `fr`—French • `it`—Italian • `ja`—Japanese • `ko`—Korean • `nl_NL`—Dutch • `no`—Norwegian

Field	Details
	• `pt_BR`—Portuguese (Brazil) • `ru`—Russian • `sv`—Swedish • `th`—Thai • `zh_CN`—Chinese (Simplified) • `zh_TW`—Chinese (Traditional)
`ManageableState`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** **Type** ManageableState enumerated list **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** Indicates the manageable state of the specified component that is contained in a package: • `beta` • `deleted` • `deprecated` • `deprecatedEditable` • `installed` • `installedEditable` • `released` • `unmanaged`
`MasterLabel`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** Label assigned to the ProductSpecificationRecType object.
`NamespacePrefix`	**Type** string **Properties** Filter, Group, Nillable, Sort

Field	Details
	Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the ***namespacePrefix__componentName*** notation. The namespace prefix can have one of the following values: • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are not Developer Edition organizations, `NamespacePrefix` is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.
`ProductSpecificationType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. Product specification type that's associated with the record type. This field is unique within your organization. The picklist values that are available to you depend on your industry solution and permission sets.
`RecordType`	**Type** sObject **Properties** Nillable **Description** Custom record type of Product2 object.
`RecordTypeId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** Required. ID of the product record type associated with the product specification type. This field is unique within your organization. This field is a relationship field.

Field	Details
	Relationship Name RecordType **Relationship Type** Lookup **Refers To** RecordType

Product Discovery

Product Discovery provides a hierarchical catalog browsing experience to identify suitable products with text-based and faceted search.

EDITIONS

Available in: Lightning Experience

Available in: **Enterprise**, **Unlimited**, and **Developer** Editions of Revenue Cloud

Product Discovery Business APIs

Use the Product Discovery Business APIs, which are composite APIs, to search products or to discover catalogs, products, and categories during the product browsing experience.

Product Discovery Standard Invocable Actions

Use the standard invocable actions available with Product Discovery to find and retrieve the product details.

Product Discovery Metadata API Types

Metadata API enables you to access some types and feature settings that you can customize in the user interface.

SEE ALSO:

Salesforce Help: Configure Product Discovery Settings

Salesforce Help: Invoke Qualification Procedures During Product Discovery

Product Discovery Business APIs

Use the Product Discovery Business APIs, which are composite APIs, to search products or to discover catalogs, products, and categories during the product browsing experience.

This table lists the available Product Discovery resources.

Resource	Description
`/connect/cpq/catalogs/`***`catalogId`*** (POST)	Get catalog details for a specified catalog ID. This API is a composite API for Product Discovery.
`/connect/cpq/catalogs` (POST)	Get a paginated list of catalogs. This API is a composite API for Product Discovery.

Resource	Description
`/connect/cpq/categories` (POST)	Get a list of categories and subcategories of a specified catalog. This API is a composite API for Product Discovery.
`/connect/cpq/categories/`***`categoryId`*** (POST)	Get details of a category for a specified category ID. This API is a composite API for Product Discovery.
`/connect/cpq/qualification` (POST)	Run the qualification procedure on a list of product IDs. This API is a composite API for Product Discovery.
`/connect/cpq/products/search` (POST)	Retrieves a list of products based on a search query or search term. This API is a composite API for Product Discovery.
`/connect/cpq/products/`***`productId`*** (POST)	Get product details, such as attributes, hierarchy, or cardinality, for a specified product ID. This API is a composite API for Product Discovery.
`/connect/cpq/products` (POST)	Get a list of products for a specified catalog, category, or subcategory. This API is a composite API for Product Discovery.
`/connect/cpq/products/bulk` (POST)	Retrieve details for multiple products. This API is a composite API for Product Discovery.
`/connect/cpq/products/guided-selection` (POST)	Retrieve a list of products based on the response identifier or search terms of a guided selection. Guided selection captures user requirements to show suitable products.

Resources

Learn more about the available Product Discovery API resources.

Request Bodies

Learn more about the available Product Discovery API request bodies.

Response Bodies

Learn more about the available Product Discovery API response bodies.

Resources

Learn more about the available Product Discovery API resources.

Catalog Details (POST)

Get catalog details for a specified catalog ID. This API is a composite API for Product Discovery.

Catalog List (POST)

Get a paginated list of catalogs. This API is a composite API for Product Discovery.

Categories List (POST)

Get a list of categories and subcategories of a specified catalog. This API is a composite API for Product Discovery.

Category Details (POST)

Get details of a category for a specified category ID. This API is a composite API for Product Discovery.

CPQ Base Bulk Details (POST)

Retrieve details for multiple products. This API is a composite API for Product Discovery.

Global Search (POST)

Retrieves a list of products based on a search query or search term. This API is a composite API for Product Discovery.

Guided Selection (POST)

Retrieve a list of products based on the response identifier or search terms of a guided selection. Guided selection captures user requirements to show suitable products.

Product Details (POST)

Get product details, such as attributes, hierarchy, or cardinality, for a specified product ID. This API is a composite API for Product Discovery.

Products List (POST)

Get a list of products for a specified catalog, category, or subcategory. This API is a composite API for Product Discovery.

Qualification (POST)

Run the qualification procedure on a list of product IDs. This API is a composite API for Product Discovery.

Catalog Details (POST)

Get catalog details for a specified catalog ID. This API is a composite API for Product Discovery.

Resource

```
/connect/cpq/catalogs/catalogId
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/cpq/catalogs/0ZSxx000000009hGAA
```

Available version

60.0

HTTP methods

POST

Request body for POST

JSON example

```
{
  "correlationId": "9cbb9650-48c5-11ed-96d1-0afcf185843b",
  "userContext": {
     "accountId": "001xx0000000001AAA",
     "contactId": "003xx00000000D7AAI"
  }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`correlationId`	String	Unique identifier value that's attached to the requests and messages, and accepts references to a particular transaction or event chain.	Optional	60.0
`userContext`	User Context Input	User context details. For example, account ID or contact ID.	Optional	60.0

Response body for POST

CPQ Base Details

Catalog List (POST)

Get a paginated list of catalogs. This API is a composite API for Product Discovery.

Resource

```
/connect/cpq/catalogs
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/cpq/catalogs
```

Available version

60.0

HTTP methods

POST

Request body for POST

JSON example

```
{
"correlationId": "9cbb9650-48c5-11ed-96d1-0afcf185843b",
"limit": 10,
```

```
    "offset": 0,
    "orderBy": [
        "name:asc",
        "id:desc"
    ],
    "userContext": {
        "accountId": "001xx0000000001AAA",
        "contactId": "003xx00000000D7AAI"
    }
    }
```

Properties

Name	Type	Description	Required or Optional	Available Version
`correlationId`	String	Unique identifier value that's attached to the requests and messages, and accepts references to a particular transaction or event chain.	Optional	60.0
`limit`	Integer	Number of items to include in the response.	Optional	60.0
`offset`	Integer	Offset size from which to get the catalog count.	Optional	60.0
`orderBy`	String[]	Sort order for the catalogs.	Optional	60.0
`userContext`	User Context Input	User context details. For example, account ID or contact ID.	Optional	60.0

Response body for POST

CPQ Base List

Categories List (POST)

Get a list of categories and subcategories of a specified catalog. This API is a composite API for Product Discovery.

Resource

```
/connect/cpq/categories
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/cpq/categories
```

Available version

60.0

HTTP methods

POST

Request body for POST

JSON example

```
{
  "correlationId": "9cbb9650-48c5-11ed-96d1-0afcf185843b",
  "catalogId": "0ZSxx000000009hGAA",
  "userContext": {
      "accountId": "001xx0000000001AAA",
      "contactId": "003xx00000000D7AAI"
    }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`catalogId`	String	ID of the catalog.	Required	60.0
`correlationId`	String	Unique identifier value that's attached to the requests and messages, and accepts references to a particular transaction or event chain.	Optional	60.0
`userContext`	User Context Input	User context details. For example, account ID or contact ID.	Optional	60.0

Response body for POST

CPQ Base List

Category Details (POST)

Get details of a category for a specified category ID. This API is a composite API for Product Discovery.

Resource

```
/connect/cpq/categories/categoryId
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/cpq/categories/0ZGxx000000001dGAA
```

Available version

60.0

HTTP methods

POST

Request body for POST

JSON example

```
{
  "correlationId": "9cbb9650-48c5-11ed-96d1-0afcf185843b",
  "userContext": {
```

```
      "accountId": "001xx0000000001AAA",
      "contactId": "003xx00000000D7AAI"
  }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
correlationId	String	Unique identifier value that's attached to the requests and messages, and accepts references to a particular transaction or event chain.	Optional	60.0
userContext	User Context Input	User context details. For example, account ID or contact ID.	Optional	60.0

Response body for POST

CPQ Base Details

CPQ Base Bulk Details (POST)

Retrieve details for multiple products. This API is a composite API for Product Discovery.

Resource

```
/connect/cpq/products/bulk
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/cpq/products/bulk
```

Available version

61.0

HTTP methods

POST

Request body for POST

JSON example

```
{
  "productData": [
    {
      "productId": "01txx0000006ivJAAQ",
      "productSellingModelId": "0jPxx000000009hEAA"
    },
    {
      "productId": "01txx0000006ivLAAQ",
      "productSellingModelId": "0jPxx000000009iEAABB"
    }
  ],
  "correlationId": "de9a674c-1807-438c-ac78-2c96f4655325",
  "priceBookId" : "01sxx0000005qxxAAA",
```

```
  "userContext": {
      "accountId": "001xx0000000001AAA",
      "contactId": "003xx00000000D7AAI"
  }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`additionalContextData`	Context Data Input[]	Additional nodes to add to the custom or default context definition. This data is appended to the context input and sent for hydration and qualification. The maximum limit of supported nodes is 10.	Optional	61.0
`additionalFields`	Map<String, Additional Fields Input>	Additional standard or custom fields of the Product2 object to include in the response. The field values are returned in the response for each of the products.	Optional	61.0
`contextDefinition`	String	Name of the custom context definition that's sent for the context creation. If unspecified, the default context definition is used.	Optional	61.0
`contextMapping`	String	Context mapping details from the context definition. If specified, the API validates if the context mapping belongs to the specified context definition and considers the mapping for hydration. If unspecified, the default context mapping of the context definition is used.	Optional	61.0
`correlationId`	String	Unique token to track and associate related events or transactions across different components of the application. If unspecified, a Universally Unique Identifier (UUID) is generated.	Optional	61.0
`currencyCode`	String	Currency code to consider for pricing and filtering.	Optional	61.0
`enablePricing`	Boolean	Indicates whether to enable pricing for the products (`true`) or not (`false`). The default value is `true`. The **Pricing Procedure** toggle from the Product Discovery Settings page from Setup overrides this property. For example, if the **Pricing Procedure** toggle	Optional	61.0

Name	Type	Description	Required or Optional	Available Version
		is disabled, then setting the `enablePricing` property to `true` has no effect and the `prices` property in the API response is returned empty.		
`enableQualification`	Boolean	Indicates whether to enable qualification rules for the products (`true`) or not (`false`). The default value is `true`. The **Qualification Procedure** toggle from the Product Discovery Settings page from Setup overrides this property. For example, if the **Qualification Procedure** toggle is disabled, then setting the `enableQualification` property to `true` has no effect and the `qualificationContext` property in the API response isn't returned.	Optional	61.0
`priceBookId`	String	ID of the price book to fetch the prices from.	Optional	61.0
`pricingProcedure`	String	Name of the custom pricing procedure to send for processing. If unspecified, the default pricing procedure is executed.	Optional	61.0
`productData`	Product Data Input[]	List of maps that contain product IDs and product selling model IDs.	Required	61.0
`qualificationProcedure`	String	Name of the custom qualification procedure to send for processing. If unspecified, the default qualification procedure is executed.	Optional	61.0
`userContext`	User Context Input[]	User context details. For example, account ID or contact ID.	Optional	61.0

Response body for POST

CPQ Base Bulk Details

Global Search (POST)

Retrieves a list of products based on a search query or search term. This API is a composite API for Product Discovery.

Resource

```
/connect/cpq/products/search
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/cpq/products/search
```

Available version

60.0

HTTP methods

POST

Request body for POST

JSON example

This example shows a sample request to search products by using a query.

```
{
    "query": {
        "textQuery": {
            "searchPhrase": "firstproduct"
        }
    },
    "catalogId": "0ZSxx0000000001GAA",
    "categoryId": "0ZGT100000000qlOAA",
    "correlationId":"9cbb9650-48c5-11ed-96d1-0afcf185843b",
    "limit":10,
    "cursor": "MTAwMDAwMDAwNg==",
    "orderBy":["name:asc","id:desc"],
    "userContext": {
      "accountId": "001xx0000000001AAA",
      "contactId": "003xx00000000D7AAI"
    },
  "additionalFields": {
    "Product2": {
       "fields" : [
            "CustomField1__c",
            "CustomField2__c",
            "StandardField1"
        ]
    }
  }
}
```

This example shows a sample request to search products by using the `searchTerm` property.

```
  {
    "searchTerm": "Laptop",
    "catalogId": "0ZSDU0000002Og64AE",
    "categoryId": "0ZGDU0000002P0A4AU",
    "correlationId":"d9d8f898-19f5-464a-ba2b-6a070783f6c4",
    "limit":10,
    "cursor": "MTAwMDAwMDAwNw==",
    "orderBy":["name:asc","id:desc"],
    "userContext": {
        "accountId": "001DU000001o2V0YAI"
    }
}
```

If a parent category ID is specified in the request body, then the API returns all products associated to all child categories.

Properties

Name	Type	Description	Required or Optional	Available Version
`additionalContextData`	Context Data Input[]	Additional nodes that are added to the custom or default context definition. The maximum number of supported nodes is 10.	Optional	60.0
`additionalFields`	Map<String, Additional Fields Input>	Additional standard or custom fields of the Product2 object to include in the response. If the requested fields are invalid or access to fields isn't available, then the API throws an error.	Optional	61.0
`catalogId`	String	ID of the catalog. If the catalog ID is specified, then the API returns the list of offers from the catalog with the pricing details related to the catalog.	Optional	60.0
`categoryId`	String	ID of the category. If the category ID isn't specified, then the API returns the matching query offers from the catalog.	Optional	60.0
`contextDefinition`	String	API name of the custom context definition that's sent for context creation. If this property isn't specified, then the default context definition is used.	Optional	60.0
`contextMapping`	String	Default context mapping of the context definition. If a context mapping is specified, then the API checks whether the mapping belongs to the specified context definition to process the details for hydration.	Optional	60.0
`correlationId`	String	Unique identifier of the request.	Optional	60.0
`currencyCode`	String	Currency code that's considered for pricing and filtering request.	Optional	60.0
`cursor`	String	Unique ID to represent the position of each product in the dataset.	Optional	60.0

Name	Type	Description	Required or Optional	Available Version
`enablePricing`	Boolean	Indicates whether to enable pricing for the products (`true`) or not (`false`). The default value is `true`. The **Pricing Procedure** toggle from the Product Discovery Settings page from Setup overrides this property. For example, if the **Pricing Procedure** toggle is disabled, then setting the `enablePricing` property to `true` has no effect and the `prices` property in the API response is returned empty.	Optional	60.0
`enableQualification`	Boolean	Indicates whether to enable qualification rules for the products (`true`) or not (`false`). The default value is `true`. The **Qualification Procedure** toggle from the Product Discovery Settings page from Setup overrides this property. For example, if the **Qualification Procedure** toggle is disabled, then setting the `enableQualification` property to `true` has no effect and the `qualificationContext` property in the API response isn't returned.	Optional	60.0
`filter`	Filter Input	Filters records based on supported criteria. The supported property is `name`. The supported operators are: • `eq` • `in` • `contains` • `gt`—Specifies a greater than criteria. Available from API version 63.0 and later for Number, Date, and Datetime data types only. • `lt`—Specifies a less than criteria. Available from API version 63.0 and later for Number, Date, and Datetime data types only. • `gte`—Specifies a greater than or equal to criteria. Available from API	Optional	60.0

Name	Type	Description	Required or Optional	Available Version
		version 63.0 and later for Number, Date, and Datetime data types only. • `lte`—Specifies a less than or equal to criteria. Available from API version 63.0 and later for Number, Date, and Datetime data types only. If multiple criteria are specified, then the resultant criteria are combined by using the `and` operator.		
`includeCatalogDetails`	Boolean	Indicates whether to include catalog details in the response (`true`) or not (`false`).	Optional	61.0
`limit`	Integer	Number of items to include in the response. The default value is 10.	Optional	60.0
`offset`	Integer	Reserved for internal use.	Optional	60.0
`orderBy`	String[]	Sort order of the results, which is either ascending or descending order. The default sort order is ascending order. The default value is `asc`.	Optional	60.0
`priceBookId`	String	ID of the price book to get the prices from. If this property isn't specified, then prices from the standard price book are fetched.	Optional	60.0
`pricingProcedure`	String	API name of the custom pricing procedure that's used for the pricing process. If this property isn't specified, then the default pricing procedure is executed.	Optional	60.0
`productClassificationId`	String	ID of the product classification.	Optional	60.0
`qualificationProcedure`	String	API name of the custom qualification procedure that's used for the product qualification process. If this property isn't specified, then the default qualification procedure is executed.	Optional	60.0
`query`	Map<String, Object>>	Query to search the products.	Required	60.0

Name	Type	Description	Required or Optional	Available Version
`relatedObjectFilter`	Related Object Filter Input[]	Filter records based on supported criteria for related objects. The supported object is `ProductSpecificationRecType`. The supported property is `IsCommerical`. The supported operator is `eq`. The supported values are `true` and `false`.	Optional	60.0
`searchTerm`	String	String used to get products with the product name containing the search term. See Search Considerations When Using Indexed Data.	Optional	62.0
`userContext`	User Context Input	User context details. For example, account ID or contact ID.	Optional	60.0

Response body for POST

CPQ Base List

Guided Selection (POST)

Retrieve a list of products based on the response identifier or search terms of a guided selection. Guided selection captures user requirements to show suitable products.

Resource

```
/connect/cpq/products/guided-selection
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/cpq/products/guided-selection
```

Available version

62.0

HTTP methods

POST

Request body for POST

JSON example

This input request includes properties from the Products List API. To learn about the other input properties, see Products List (POST).

```
{
  "correlationId": "corrId",
  "catalogId": "0ZSxx0000000001GAA",
```

```
  "priceBookId": "pricebookId",
  "limit": 10,
  "cursor": "MTAwMDAwMDAwNg==",
  "userContext": {
    "accountId": "accId"
  },
  "guidedSelectionResponseId": "ABCxx0000000001GAA",
  "searchTerms": [
    {
      "term": "IPhone"
    },
    {
      "term": "4GB"
    },
    {
      "term": "64GB"
    }
  ],
  "enableQualification": true,
  "enablePricing": true,
  "includeCatalogDetails": false
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`guidedSelectionResponseId`	String	Response identifier of the guided selection.	Required if the `searchTerms` property isn't specified.	62.0
`searchTerms`	Guided Selection Search Term Input[]	Search terms of the guided selection.	Required if the `guidedSelectionResponseId` property isn't specified.	62.0

If both the `guidedSelectionResponseId` and `searchTerms` properties are specified, then the `searchTerms` property is considered in the input request.

Response body for POST

Guided Selection

Product Details (POST)

Get product details, such as attributes, hierarchy, or cardinality, for a specified product ID. This API is a composite API for Product Discovery.

Resource

```
/connect/cpq/products/productId
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/cpq/products/01txx0000006j08AAA
```

Available version

60.0

HTTP methods

POST

Request body for POST

JSON example

```
{
  "correlationId": "9cbb9650-48c5-11ed-96d1-0afcf185843b",
  "catalogId": "0ZSxx0000000001GAA",
  "priceBookId" : "01s26000002ZT71AAG",
  "productSellingModelId" : "0jP1Q000000CaVFUA0",
    "userContext": {
      "accountId": "001xx0000000001AAA",
      "contactId": "003xx00000000D7AAI"
  },
  "enablePricing" : true,
  "enableQualification" : true,
  "qualificationProcedure": "QualificationProcedure",
  "pricingProcedure": "Preview",
  "contextDefinition": "TestDefinition",
  "contextMapping": "TestDefinitionNode",
  "additionalFields": {
    "Product2": {
      "fields": [
        "field1",
        "field2"
      ]
    },
    "ProductAttributeDefinition": {
      "fields": [
        "field3",
        "field4"
      ]
    }
  },
  "additionalContextData":
 [{
    "nodeName": "Contract",
    "nodeData":
  {
    "id": "xxxxx231",
    "name": "Contract1"
  }
   },
  {
    "nodeName": "Lead",
    "nodeData":
  {
```

```
    "id": "111111131",
    "name": "Lead1"
  }
 }]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
additionalContextData	Context Data Input[]	Additional nodes that are added to the custom or default context definition. The maximum number of supported nodes is 10.	Optional	60.0
additionalFields	Map<String, Additional Fields Input>	Additional standard or custom fields to include in the response. The supported objects are: • Product2 • ProductAttributeDefinition—If the fields defined for the ProductAttributeDefinition object aren't available for the ProductClassificationAttr object, then the API request fails.	Optional	61.0
catalogId	String	ID of the catalog. If the catalog ID is specified, then the API returns the list of offers from the catalog with the pricing details related to the catalog.	Optional	60.0
contextDefinition	String	API name of the custom context definition that's sent for context creation. If this property isn't specified, then the default context definition is used.	Optional	60.0
contextMapping	String	Default context mapping of the context definition. If a context mapping is specified, then the API checks whether the mapping belongs to the specified context definition to process the details for hydration.	Optional	60.0
correlationId	String	Unique identifier value that's attached to the requests and messages, and accepts references to a particular transaction or event chain.	Optional	60.0
currencyCode	String	Currency code that's considered for pricing and filtering request. If multiple currencies are enabled for the org, then	Optional	60.0

Name	Type	Description	Required or Optional	Available Version
		the `currencyCode` property is required.		
`enable Pricing`	Boolean	Indicates whether to enable pricing for the products (`true`) or not (`false`). The default value is `true`. The **Pricing Procedure** toggle from the Product Discovery Settings page from Setup overrides this property. For example, if the **Pricing Procedure** toggle is disabled, then setting the `enablePricing` property to `true` has no effect and the `prices` property in the API response is returned empty.	Optional	60.0
`enable Qualification`	Boolean	Indicates whether to enable qualification rules for the products (`true`) or not (`false`). The default value is `true`. The **Qualification Procedure** toggle from the Product Discovery Settings page from Setup overrides this property. For example, if the **Qualification Procedure** toggle is disabled, then setting the `enableQualification` property to `true` has no effect and the `qualificationContext` property in the API response isn't returned.	Optional	60.0
`priceBookId`	String	ID of the price book to fetch the prices from. If this property isn't specified, then the prices from the standard price book are fetched.	Required	60.0
`pricing Procedure`	String	API name of the custom pricing procedure that's used for the pricing process. If this property isn't specified, then the default pricing procedure is executed.	Optional	60.0
`product SellingModel Id`	String	ID of the product selling model.	Optional	60.0
`qualification Procedure`	String	API name of the custom qualification procedure that's used for the product qualification process. If this property isn't	Optional	60.0

Name	Type	Description	Required or Optional	Available Version
		specified, then the default qualification procedure is executed.		
`userContext`	User Context Input	User context details. For example, account ID or contact ID.	Optional	60.0

Response body for POST

CPQ Base Details

Products List (POST)

Get a list of products for a specified catalog, category, or subcategory. This API is a composite API for Product Discovery.

Resource

```
/connect/cpq/products
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/cpq/products
```

Available version

60.0

HTTP methods

POST

Request body for POST

JSON example

```
{
  "correlationId": "eeaa1db2-f371-4227-a886-c77e2f66ce1d",
  "limit": 60,
  "cursor": "MTAwMDAwMDAwNg==",
  "orderBy": [
    "name:asc"
  ],
  "catalogId": "0ZSDU0000002Og74AE",
  "categoryId": "0ZGDU0000002P0H4AU",
  "priceBookId": "01sDU000000JVsVYAW",
  "productClassificationId": "11BDU0000002TCC2A2",
  "currencyCode": "USD",
  "userContext": {
    "accountId": "001DU000001o2UzYAI"
  },
  "includeCatalogDetails": true,
  "enableQualification": true,
  "enablePricing": true,
  "qualificationProcedure": "ProductQualification",
  "pricingProcedure": "pricingProcedure",
  "contextDefinition": "BrowseContextDefinitionExt",
  "contextMapping": "ProductDiscoveryMapping",
```

```
  "filter": {
    "criteria": [
      {
        "property": "name",
        "operator": "eq",
        "value": "Laptop Pro Bundle"
      }
    ]
  },
  "relatedObjectFilters": [
    {
      "objectName": "ProductSpecificationRecType",
      "criteria": [
        {
          "property": "IsCommercial",
          "operator": "eq",
          "value": true
        }
      ]
    }
  ],
  "additionalContextData": [
    {
      "nodeName": "Account",
      "nodeData": {
        "id": "001DU000001o2UzYAI",
        "name": "Cloud Kicks"
      }
    }
  ],
  "additionalFields": {
    "Product2": {
      "fields": [
        "CanRamp",
        "DecompositionScope",
        "ProductCode"
      ]
    }
  }
}
```

If a parent category ID is specified in the request body, then the API returns all products associated to all child categories.

Properties

Name	Type	Description	Required or Optional	Available Version
`additionalContextData`	Context Data Input[]	Additional nodes that are added to the custom or default context definition. The maximum number of supported nodes is 10.	Optional	60.0

Name	Type	Description	Required or Optional	Available Version
`additionalFields`	Map<String, Additional Fields Input>	Additional standard or custom fields of the Product2 object to include in the response. If the requested fields are invalid or access to fields isn't available, then the API throws an error.	Optional	61.0
`catalogId`	String	ID of the catalog. If the catalog ID is specified, then the API returns the list of offers from the catalog with the pricing details related to the catalog.	Optional	60.0
`categoryId`	String	ID of the category. If the category ID isn't specified, then the API returns the list of offers from the catalog.	Optional	60.0
`contextDefinition`	String	API name of the custom context definition that's sent for context creation. If this property isn't specified, then the default context definition is used.	Optional	60.0
`contextMapping`	String	Default context mapping of the context definition. If a context mapping is specified, then the API checks whether the mapping belongs to the specified context definition to process the details for hydration.	Optional	60.0
`correlationId`	String	Unique identifier of the request.	Optional	60.0
`currencyCode`	String	Currency code that's considered for pricing and filtering request. If multiple currencies are enabled for the org, then the `currencyCode` property is required.	Optional	60.0
`cursor`	String	Unique ID to represent the position of each product in the dataset.	Optional	60.0
`enablePricing`	Boolean	Indicates whether to enable pricing for the products (`true`) or not (`false`). The default value is `true`. The **Pricing Procedure** toggle from the Product Discovery Settings page from Setup overrides this property. For example, if the **Pricing Procedure** toggle is disabled, then setting the	Optional	60.0

Name	Type	Description	Required or Optional	Available Version
		`enablePricing` property to `true` has no effect and the `prices` property in the API response is returned empty.		
`enableQualification`	Boolean	Indicates whether to enable qualification rules for the products (`true`) or not (`false`). The default value is `true`. The **Qualification Procedure** toggle from the Product Discovery Settings page from Setup overrides this property. For example, if the **Qualification Procedure** toggle is disabled, then setting the `enableQualification` property to `true` has no effect and the `qualificationContext` property in the API response isn't returned.	Optional	60.0
`filter`	Filter Input	Filters records based on supported criteria. The supported property is `name`. The supported operators are: • `eq` • `in` • `contains`—This value isn't applicable if the **Use Indexed Data For Product Listing and Search** toggle from the Product Discovery Settings page from Setup is enabled. If multiple criteria are specified, then the resultant criteria are combined by using the `and` operator.	Optional	60.0
`includeCatalogDetails`	Boolean	Indicates whether to include catalog details in the response (`true`) or not (`false`).	Optional	61.0
`limit`	Integer	Number of items to include in the response. The default value is 10.	Optional	60.0
`offset`	Integer	Reserved for internal use.	Optional	60.0
`orderBy`	String[]	Sort order of the results, which is either ascending (`asc`) or descending order (`desc`). The default sort order is	Optional	60.0

Name	Type	Description	Required or Optional	Available Version
		ascending order. The default value is `asc`. If the **Use Indexed Data For Product Listing and Search** toggle from the Product Discovery Settings page from Setup is enabled, then you can sort products by using name only.		
`priceBookId`	String	ID of the price book to get the prices from. If this property isn't specified, then prices from the standard price book are fetched.	Required	60.0
`pricingProcedure`	String	API name of the custom pricing procedure that's used for the pricing process. If this property isn't specified, then the default pricing procedure is executed.	Optional	60.0
`productClassificationId`	String	ID of the product classification.	Optional	60.0
`qualificationProcedure`	String	API name of the custom qualification procedure that's used for the product qualification process. If this property isn't specified, then the default qualification procedure is executed.	Optional	60.0
`relatedObjectFilters`	Related Object Filter Input[]	Filter records based on supported criteria for related objects. The supported object is `ProductSpecificationRecType`. The supported property is `IsCommerical`. The supported operator is `eq`. The supported values are `true` and `false`.	Optional	60.0
`userContext`	User Context Input	User context details. For example, account ID or contact ID.	Optional	60.0

Response body for POST

CPQ Base List

Qualification (POST)

Run the qualification procedure on a list of product IDs. This API is a composite API for Product Discovery.

Resource

```
/connect/cpq/qualification
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/cpq/qualification
```

Available version

60.0

HTTP methods

POST

Request body for POST

JSON example

```
{
  "productIds": [
    "01txx0000006i7PAAQ",
    "01txx0000006i7QAAQ",
    "01txx0000006i7IAAQ"
  ],
  "userContext": {
    "accountId": "001xx000003GZHgAAO"
  }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
additionalContextData	Context Data Input[]	Additional nodes that are added to the custom or default context definition. The maximum number of supported nodes is 10.	Optional	60.0
catalogId	String	ID of the catalog.	Optional	60.0
categoryId	String	ID of the category.	Optional	60.0
contextDefinition	String	API name of the custom context definition that's sent for context creation. If this property isn't specified, the default context definition is used.	Optional	60.0
contextMapping	String	Default context mapping of the context definition. If a context mapping is specified, then the API checks whether the mapping belongs to the specified context definition to process the details for hydration.	Optional	60.0

Name	Type	Description	Required or Optional	Available Version
correlationId	String	Unique identifier of the request.	Optional	60.0
productIds	String[]	List of product IDs for qualification check.	Required	60.0
qualificationProcedure	String	API name of the custom qualification procedure that's sent for processing. If this property isn't specified, the default qualification procedure is executed.	Optional	60.0
userContext	User Context Input	User context details. For example, account ID or contact ID.	Optional	60.0

Response body for POST

CPQ Base List

Request Bodies

Learn more about the available Product Discovery API request bodies.

Additional Fields Input

Input representation of the additional standard or custom fields to include in the response.

Bulk Product Details Input

Input representation of the request to retrieve details of multiple products.

Catalog Details Input

Input representation of the request to get the catalog details.

Catalog List Input

Input representation of the request to get a list of catalogs.

Category Details Input

Input representation of the request to get category details.

Category List Input

Input representation of the request to get a list of categories.

Context Data Input

Input representation of the context data.

Guided Selection Input

Input representation of the guided selection details.

Guided Selection Search Term Input

Input representation of the search terms of a guided selection.

Filter Criteria Input

Input representation of the criteria to filter records based on supported properties.

Filter Input

Input representation of the request to filter records.

Product Details Input
Input representation of the request to get product details.

Product Data Input
Input representation of the product details such as the product ID and product selling model ID.

Product List Input
Input representation of the request to retrieve a list of products.

Products Search Input
Input representation of the request to search products.

QOC Qualification
Input representation of the qualification request.

Related Object Filter Input
Input representation of the request to filter records of a related object.

User Context Input
Input representation of the details with the user context.

Additional Fields Input

Input representation of the additional standard or custom fields to include in the response.

JSON example

```
  "additionalFields" : {
    "Product2" : {
    "fields" : ["CustomField1__c","StandardField1"]
}
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`fields`	String[]	List of additional standard or custom fields to include in the response.	Optional	61.0

Bulk Product Details Input

Input representation of the request to retrieve details of multiple products.

JSON example

```
{
  "productData": [
    {
      "productId": "01txx0000006ivJAAQ",
      "productSellingModelId": "0jPxx000000009hEAA"
    },
    {
      "productId": "01txx0000006ivLAAQ",
      "productSellingModelId": "0jPxx000000009iEAABB"
```

```
    }
  ],
  "correlationId": "de9a674c-1807-438c-ac78-2c96f4655325",
  "priceBookId" : "01sxx0000005qxxAAA",
  "userContext": {
      "accountId": "001xx0000000001AAA",
      "contactId": "003xx00000000D7AAI"
  }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`additionalContextData`	Context Data Input[]	Additional nodes to add to the custom or default context definition. This data is appended to the context input and sent for hydration and qualification. The maximum limit of supported nodes is 10.	Optional	61.0
`additionalFields`	Map<String, Additional Fields Input>	Additional standard or custom fields of the Product2 object to include in the response. The field values are returned in the response for each of the products.	Optional	61.0
`contextDefinition`	String	Name of the custom context definition that's sent for the context creation. If unspecified, the default context definition is used.	Optional	61.0
`contextMapping`	String	Context mapping details from the context definition. If specified, the API validates if the context mapping belongs to the specified context definition and considers the mapping for hydration. If unspecified, the default context mapping of the context definition is used.	Optional	61.0
`correlationId`	String	Unique token to track and associate related events or transactions across different components of the application. If unspecified, a Universally Unique Identifier (UUID) is generated.	Optional	61.0
`currencyCode`	String	Currency code to consider for pricing and filtering.	Optional	61.0

Name	Type	Description	Required or Optional	Available Version
`enablePricing`	Boolean	Indicates whether to enable pricing for the products (`true`) or not (`false`). The default value is `true`. The **Pricing Procedure** toggle from the Product Discovery Settings page from Setup overrides this property. For example, if the **Pricing Procedure** toggle is disabled, then setting the `enablePricing` property to `true` has no effect and the `prices` property in the API response is returned empty.	Optional	61.0
`enable Qualification`	Boolean	Indicates whether to enable qualification rules for the products (`true`) or not (`false`). The default value is `true`. The **Qualification Procedure** toggle from the Product Discovery Settings page from Setup overrides this property. For example, if the **Qualification Procedure** toggle is disabled, then setting the `enableQualification` property to `true` has no effect and the `qualificationContext` property in the API response isn't returned.	Optional	61.0
`priceBookId`	String	ID of the price book to fetch the prices from.	Optional	61.0
`pricing Procedure`	String	Name of the custom pricing procedure to send for processing. If unspecified, the default pricing procedure is executed.	Optional	61.0
`productData`	Product Data Input[]	List of maps that contain product IDs and product selling model IDs.	Required	61.0
`qualification Procedure`	String	Name of the custom qualification procedure to send for processing. If unspecified, the default qualification procedure is executed.	Optional	61.0
`userContext`	User Context Input[]	User context details. For example, account ID or contact ID.	Optional	61.0

Catalog Details Input

Input representation of the request to get the catalog details.

JSON example

```
{
  "correlationId": "9cbb9650-48c5-11ed-96d1-0afcf185843b",
  "userContext": {
     "accountId": "001xx0000000001AAA",
     "contactId": "003xx00000000D7AAI"
  }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`correlationId`	String	Unique identifier value that's attached to the requests and messages, and accepts references to a particular transaction or event chain.	Optional	60.0
`userContext`	User Context Input	User context details. For example, account ID or contact ID.	Optional	60.0

Catalog List Input

Input representation of the request to get a list of catalogs.

JSON example

```
    {
    "correlationId": "9cbb9650-48c5-11ed-96d1-0afcf185843b",
    "limit": 10,
    "offset": 0,
    "orderBy": [
        "name:asc",
        "id:desc"
    ],
    "userContext": {
        "accountId": "001xx0000000001AAA",
        "contactId": "003xx00000000D7AAI"
    }
    }
```

Properties

Name	Type	Description	Required or Optional	Available Version
`correlationId`	String	Unique identifier value that's attached to the requests and messages, and accepts references to a particular transaction or event chain.	Optional	60.0

Name	Type	Description	Required or Optional	Available Version
limit	Integer	Number of items to include in the response.	Optional	60.0
offset	Integer	Offset size from which to get the catalog count.	Optional	60.0
orderBy	String[]	Sort order for the catalogs.	Optional	60.0
userContext	User Context Input	User context details. For example, account ID or contact ID.	Optional	60.0

Category Details Input

Input representation of the request to get category details.

JSON example

```
{
  "correlationId": "9cbb9650-48c5-11ed-96d1-0afcf185843b",
  "userContext": {
      "accountId": "001xx0000000001AAA",
      "contactId": "003xx00000000D7AAI"
  }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
correlationId	String	Unique identifier value that's attached to the requests and messages, and accepts references to a particular transaction or event chain.	Optional	60.0
userContext	User Context Input	User context details. For example, account ID or contact ID.	Optional	60.0

Category List Input

Input representation of the request to get a list of categories.

JSON example

```
{
  "correlationId": "9cbb9650-48c5-11ed-96d1-0afcf185843b",
  "catalogId": "0ZSxx000000009hGAA",
  "userContext": {
      "accountId": "001xx0000000001AAA",
      "contactId": "003xx00000000D7AAI"
```

```
    }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
catalogId	String	ID of the catalog.	Required	60.0
correlationId	String	Unique identifier value that's attached to the requests and messages, and accepts references to a particular transaction or event chain.	Optional	60.0
userContext	User Context Input	User context details. For example, account ID or contact ID.	Optional	60.0

Context Data Input

Input representation of the context data.

JSON example

```
"additionalContextData":[
  {
  "nodeName": "Contract",
  "nodeData": {
    "id": "xxxxx231",
    "name": "Contract1"

  }
},
{
  "nodeName": "Lead",
  "nodeData": {
    "id": "lllllll31",
    "name": "Lead1"

  }
}]
```

Properties

Name	Type	Description	Required or Optional	Available Version
nodeData	Map<String, Object>	Details of the node.	Optional	60.0
nodeName	String	Name of the node.	Optional	60.0

Guided Selection Input

Input representation of the guided selection details.

JSON example

This input request includes properties from the Products List API. To learn about the other input properties, see Products List (POST).

```
{
  "correlationId": "corrId",
  "catalogId": "0ZSxx0000000001GAA",
  "priceBookId": "pricebookId",
  "limit": 10,
  "cursor": "MTAwMDAwMDAwNg==",
  "userContext": {
    "accountId": "accId"
  },
  "guidedSelectionResponseId": "ABCxx0000000001GAA",
  "searchTerms": [
    {
      "term": "IPhone"
    },
    {
      "term": "4GB"
    },
    {
      "term": "64GB"
    }
  ],
  "enableQualification": true,
  "enablePricing": true,
  "includeCatalogDetails": false
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`guidedSelectionResponseId`	String	Response identifier of the guided selection.	Required if the `searchTerms` property isn't specified.	62.0
`searchTerms`	Guided Selection Search Term Input[]	Search terms of the guided selection.	Required if the `guidedSelectionResponseId` property isn't specified.	62.0

If both the `guidedSelectionResponseId` and `searchTerms` properties are specified, then the `searchTerms` property is considered in the input request.

Guided Selection Search Term Input

Input representation of the search terms of a guided selection.

JSON example

```
"searchTerms": [
  {
    "term": "IPhone"
  },
  {
    "term": "4GB"
  },
  {
    "term": "64GB"
  }
]
```

Properties

Name	Type	Description	Required or Optional	Available Version
`term`	String	Search term for the guided selection.	Required	62.0

Filter Criteria Input

Input representation of the criteria to filter records based on supported properties.

JSON example

```
"criteria":
 [{
   "attributeType": "ProductStandard",
   "property": "name",
   "operator": "eq",
   "value": "iPhone"
 }]
```

Properties

Name	Type	Description	Required or Optional	Available Version
`attributeType`	String	Search attribute type of the facet for a faceted search. Valid values are: • `ProductStandard` • `ProductCustom` • `ProductDynamicAttribute` • `ProductAttributeStandard` • `ProductAttributeCustom`	Optional	63.0

Name	Type	Description	Required or Optional	Available Version
`operator`	String	Operator used for the filter criteria. The supported operators are: • `eq` • `in` • `contains` • `gt`—Specifies a greater than criteria. Available from API version 63.0 and later for Number, Date, and Datetime data types only. • `lt`—Specifies a less than criteria. Available from API version 63.0 and later for Number, Date, and Datetime data types only. • `gte`—Specifies a greater than or equal to criteria. Available from API version 63.0 and later for Number, Date, and Datetime data types only. • `lte`—Specifies a less than or equal to criteria. Available from API version 63.0 and later for Number, Date, and Datetime data types only.	Required	60.0
`property`	String	Property name to use in the filter, which must be the same as the object field. The supported property is `name`.	Required	60.0
`value`	Object	Value for the filter criteria.	Required	60.0

Filter Input

Input representation of the request to filter records.

JSON example

```
"filter":
 {"criteria":
 [ {
   "attributeType": "ProductStandard",
   "property": "name",
   "operator": "eq",
   "value": "iPhone"
 }]}
```

Properties

Name	Type	Description	Required or Optional	Available Version
criteria	Filter Criteria Input[]	Filter criteria to filter the records.	Optional	60.0

Product Details Input

Input representation of the request to get product details.

JSON example

```
{
  "correlationId": "9cbb9650-48c5-11ed-96d1-0afcf185843b",
  "catalogId": "0ZSxx0000000001GAA",
  "priceBookId" : "01s26000002ZT71AAG",
  "productSellingModelId" : "0jP1Q000000CaVFUA0",
    "userContext": {
      "accountId": "001xx0000000001AAA",
      "contactId": "003xx00000000D7AAI"
  },
  "enablePricing" : true,
  "enableQualification" : true,
  "qualificationProcedure": "QualificationProcedure",
  "pricingProcedure": "Preview",
  "contextDefinition": "TestDefinition",
  "contextMapping": "TestDefinitionNode",
  "additionalFields": {
    "Product2": {
      "fields": [
        "field1",
        "field2"
      ]
    },
    "ProductAttributeDefinition": {
      "fields": [
        "field3",
        "field4"
      ]
    }
  },
  "additionalContextData":
 [{
    "nodeName": "Contract",
    "nodeData":
  {
    "id": "xxxxx231",
    "name": "Contract1"
  }
   },
  {
    "nodeName": "Lead",
    "nodeData":
```

```
  {
    "id": "lllllll31",
    "name": "Lead1"
  }
 }]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
additional ContextData	Context Data Input[]	Additional nodes that are added to the custom or default context definition. The maximum number of supported nodes is 10.	Optional	60.0
additional Fields	Map<String, Additional Fields Input>	Additional standard or custom fields to include in the response. The supported objects are: • Product2 • ProductAttributeDefinition—If the fields defined for the ProductAttributeDefinition object aren't available for the ProductClassificationAttr object, then the API request fails.	Optional	61.0
catalogId	String	ID of the catalog. If the catalog ID is specified, then the API returns the list of offers from the catalog with the pricing details related to the catalog.	Optional	60.0
context Definition	String	API name of the custom context definition that's sent for context creation. If this property isn't specified, then the default context definition is used.	Optional	60.0
context Mapping	String	Default context mapping of the context definition. If a context mapping is specified, then the API checks whether the mapping belongs to the specified context definition to process the details for hydration.	Optional	60.0
correlationId	String	Unique identifier value that's attached to the requests and messages, and accepts references to a particular transaction or event chain.	Optional	60.0
currencyCode	String	Currency code that's considered for pricing and filtering request. If multiple currencies	Optional	60.0

Name	Type	Description	Required or Optional	Available Version
		are enabled for the org, then the `currencyCode` property is required.		
`enablePricing`	Boolean	Indicates whether to enable pricing for the products (`true`) or not (`false`). The default value is `true`. The **Pricing Procedure** toggle from the Product Discovery Settings page from Setup overrides this property. For example, if the **Pricing Procedure** toggle is disabled, then setting the `enablePricing` property to `true` has no effect and the `prices` property in the API response is returned empty.	Optional	60.0
`enable Qualification`	Boolean	Indicates whether to enable qualification rules for the products (`true`) or not (`false`). The default value is `true`. The **Qualification Procedure** toggle from the Product Discovery Settings page from Setup overrides this property. For example, if the **Qualification Procedure** toggle is disabled, then setting the `enableQualification` property to `true` has no effect and the `qualificationContext` property in the API response isn't returned.	Optional	60.0
`priceBookId`	String	ID of the price book to fetch the prices from. If this property isn't specified, then the prices from the standard price book are fetched.	Required	60.0
`pricing Procedure`	String	API name of the custom pricing procedure that's used for the pricing process. If this property isn't specified, then the default pricing procedure is executed.	Optional	60.0
`product SellingModel Id`	String	ID of the product selling model.	Optional	60.0
`qualification Procedure`	String	API name of the custom qualification procedure that's used for the product qualification process. If this property isn't specified, then the default qualification procedure is executed.	Optional	60.0

Name	Type	Description	Required or Optional	Available Version
`userContext`	User Context Input	User context details. For example, account ID or contact ID.	Optional	60.0

Product Data Input

Input representation of the product details such as the product ID and product selling model ID.

JSON example

```
"productData": [
  {
    "productId": "01txx0000006ivJAAQ",
    "productSellingModelId": "0jPxx000000009hEAA"
  },
  {
    "productId": "01txx0000006ivLAAQ",
    "productSellingModelId": "0jPxx000000009iEAABB"
  }
]
```

Properties

Name	Type	Description	Required or Optional	Available Version
`productId`	String	ID of the product.	Required	61.0
`productSellingModelId`	String	ID of the product selling model.	Optional	61.0

Product List Input

Input representation of the request to retrieve a list of products.

JSON example

```
{
  "correlationId": "eeaa1db2-f371-4227-a886-c77e2f66ce1d",
  "limit": 60,
  "cursor": "MTAwMDAwMDAwNg==",
  "orderBy": [
    "name:asc"
  ],
  "catalogId": "0ZSDU0000002Og74AE",
  "categoryId": "0ZGDU0000002P0H4AU",
  "priceBookId": "01sDU000000JVsVYAW",
  "productClassificationId": "11BDU0000002TCC2A2",
  "currencyCode": "USD",
  "userContext": {
    "accountId": "001DU000001o2UzYAI"
```

```
  },
  "includeCatalogDetails": true,
  "enableQualification": true,
  "enablePricing": true,
  "qualificationProcedure": "ProductQualification",
  "pricingProcedure": "pricingProcedure",
  "contextDefinition": "BrowseContextDefinitionExt",
  "contextMapping": "ProductDiscoveryMapping",
  "filter": {
    "criteria": [
      {
        "property": "name",
        "operator": "eq",
        "value": "Laptop Pro Bundle"
      }
    ]
  },
  "relatedObjectFilters": [
    {
      "objectName": "ProductSpecificationRecType",
      "criteria": [
        {
          "property": "IsCommercial",
          "operator": "eq",
          "value": true
        }
      ]
    }
  ],
  "additionalContextData": [
    {
      "nodeName": "Account",
      "nodeData": {
        "id": "001DU000001o2UzYAI",
        "name": "Cloud Kicks"
      }
    }
  ],
  "additionalFields": {
    "Product2": {
      "fields": [
        "CanRamp",
        "DecompositionScope",
        "ProductCode"
      ]
    }
  }
}
```

If a parent category ID is specified in the request body, then the API returns all products associated to all child categories.

Properties

Name	Type	Description	Required or Optional	Available Version
`additionalContextData`	Context Data Input[]	Additional nodes that are added to the custom or default context definition. The maximum number of supported nodes is 10.	Optional	60.0
`additionalFields`	Map<String, Additional Fields Input>	Additional standard or custom fields of the Product2 object to include in the response. If the requested fields are invalid or access to fields isn't available, then the API throws an error.	Optional	61.0
`catalogId`	String	ID of the catalog. If the catalog ID is specified, then the API returns the list of offers from the catalog with the pricing details related to the catalog.	Optional	60.0
`categoryId`	String	ID of the category. If the category ID isn't specified, then the API returns the list of offers from the catalog.	Optional	60.0
`contextDefinition`	String	API name of the custom context definition that's sent for context creation. If this property isn't specified, then the default context definition is used.	Optional	60.0
`contextMapping`	String	Default context mapping of the context definition. If a context mapping is specified, then the API checks whether the mapping belongs to the specified context definition to process the details for hydration.	Optional	60.0
`correlationId`	String	Unique identifier of the request.	Optional	60.0
`currencyCode`	String	Currency code that's considered for pricing and filtering request. If multiple currencies are enabled for the org, then the `currencyCode` property is required.	Optional	60.0
`cursor`	String	Unique ID to represent the position of each product in the dataset.	Optional	60.0
`enablePricing`	Boolean	Indicates whether to enable pricing for the products (`true`) or not (`false`). The default value is `true`. The **Pricing Procedure** toggle from the Product Discovery Settings page from	Optional	60.0

Name	Type	Description	Required or Optional	Available Version
		Setup overrides this property. For example, if the **Pricing Procedure** toggle is disabled, then setting the `enablePricing` property to `true` has no effect and the `prices` property in the API response is returned empty.		
`enableQualification`	Boolean	Indicates whether to enable qualification rules for the products (`true`) or not (`false`). The default value is `true`. The **Qualification Procedure** toggle from the Product Discovery Settings page from Setup overrides this property. For example, if the **Qualification Procedure** toggle is disabled, then setting the `enableQualification` property to `true` has no effect and the `qualificationContext` property in the API response isn't returned.	Optional	60.0
`filter`	Filter Input	Filters records based on supported criteria. The supported property is `name`. The supported operators are: • `eq` • `in` • `contains`—This value isn't applicable if the **Use Indexed Data For Product Listing and Search** toggle from the Product Discovery Settings page from Setup is enabled. If multiple criteria are specified, then the resultant criteria are combined by using the `and` operator.	Optional	60.0
`includeCatalogDetails`	Boolean	Indicates whether to include catalog details in the response (`true`) or not (`false`).	Optional	61.0
`limit`	Integer	Number of items to include in the response. The default value is 10.	Optional	60.0
`offset`	Integer	Reserved for internal use.	Optional	60.0
`orderBy`	String[]	Sort order of the results, which is either ascending (`asc`) or descending order	Optional	60.0

Name	Type	Description	Required or Optional	Available Version
		(`desc`). The default sort order is ascending order. The default value is `asc`. If the **Use Indexed Data For Product Listing and Search** toggle from the Product Discovery Settings page from Setup is enabled, then you can sort products by using name only.		
`priceBookId`	String	ID of the price book to get the prices from. If this property isn't specified, then prices from the standard price book are fetched.	Required	60.0
`pricingProcedure`	String	API name of the custom pricing procedure that's used for the pricing process. If this property isn't specified, then the default pricing procedure is executed.	Optional	60.0
`productClassificationId`	String	ID of the product classification.	Optional	60.0
`qualificationProcedure`	String	API name of the custom qualification procedure that's used for the product qualification process. If this property isn't specified, then the default qualification procedure is executed.	Optional	60.0
`relatedObjectFilters`	Related Object Filter Input[]	Filter records based on supported criteria for related objects. The supported object is `ProductSpecificationRecType`. The supported property is `IsCommerical`. The supported operator is `eq`. The supported values are `true` and `false`.	Optional	60.0
`userContext`	User Context Input	User context details. For example, account ID or contact ID.	Optional	60.0

Products Search Input

Input representation of the request to search products.

JSON example

This example shows a sample request to search products by using a query.

```
{
    "query": {
        "textQuery": {
            "searchPhrase": "firstproduct"
        }
    },
    "catalogId": "0ZSxx0000000001GAA",
    "categoryId": "0ZGT100000000qlOAA",
    "correlationId":"9cbb9650-48c5-11ed-96d1-0afcf185843b",
    "limit":10,
    "cursor": "MTAwMDAwMDAwNg==",
    "orderBy":["name:asc","id:desc"],
    "userContext": {
      "accountId": "001xx0000000001AAA",
      "contactId": "003xx00000000D7AAI"
    },
  "additionalFields": {
    "Product2": {
       "fields" : [
            "CustomField1__c",
            "CustomField2__c",
            "StandardField1"
        ]
    }
  }
}
```

This example shows a sample request to search products by using the `searchTerm` property.

```
  {
    "searchTerm": "Laptop",
    "catalogId": "0ZSDU0000002Og64AE",
    "categoryId": "0ZGDU0000002P0A4AU",
    "correlationId":"d9d8f898-19f5-464a-ba2b-6a070783f6c4",
    "limit":10,
    "cursor": "MTAwMDAwMDAwNw==",
    "orderBy":["name:asc","id:desc"],
    "userContext": {
        "accountId": "001DU000001o2V0YAI"
    }
}
```

If a parent category ID is specified in the request body, then the API returns all products associated to all child categories.

Properties

Name	Type	Description	Required or Optional	Available Version
`additionalContextData`	Context Data Input[]	Additional nodes that are added to the custom or default context definition. The	Optional	60.0

Name	Type	Description	Required or Optional	Available Version
		maximum number of supported nodes is 10.		
`additionalFields`	Map<String, Additional Fields Input>	Additional standard or custom fields of the Product2 object to include in the response. If the requested fields are invalid or access to fields isn't available, then the API throws an error.	Optional	61.0
`catalogId`	String	ID of the catalog. If the catalog ID is specified, then the API returns the list of offers from the catalog with the pricing details related to the catalog.	Optional	60.0
`categoryId`	String	ID of the category. If the category ID isn't specified, then the API returns the matching query offers from the catalog.	Optional	60.0
`contextDefinition`	String	API name of the custom context definition that's sent for context creation. If this property isn't specified, then the default context definition is used.	Optional	60.0
`contextMapping`	String	Default context mapping of the context definition. If a context mapping is specified, then the API checks whether the mapping belongs to the specified context definition to process the details for hydration.	Optional	60.0
`correlationId`	String	Unique identifier of the request.	Optional	60.0
`currencyCode`	String	Currency code that's considered for pricing and filtering request.	Optional	60.0
`cursor`	String	Unique ID to represent the position of each product in the dataset.	Optional	60.0
`enablePricing`	Boolean	Indicates whether to enable pricing for the products (`true`) or not (`false`). The default value is `true`. The **Pricing Procedure** toggle from the Product Discovery Settings page from Setup overrides this property. For example, if the **Pricing Procedure** toggle is disabled, then setting the `enablePricing` property to `true` has no effect and the `prices` property in the API response is returned empty.	Optional	60.0

Name	Type	Description	Required or Optional	Available Version
`enableQualification`	Boolean	Indicates whether to enable qualification rules for the products (`true`) or not (`false`). The default value is `true`. The **Qualification Procedure** toggle from the Product Discovery Settings page from Setup overrides this property. For example, if the **Qualification Procedure** toggle is disabled, then setting the `enableQualification` property to `true` has no effect and the `qualificationContext` property in the API response isn't returned.	Optional	60.0
`filter`	Filter Input	Filters records based on supported criteria. The supported property is `name`. The supported operators are: • `eq` • `in` • `contains` • `gt`—Specifies a greater than criteria. Available from API version 63.0 and later for Number, Date, and Datetime data types only. • `lt`—Specifies a less than criteria. Available from API version 63.0 and later for Number, Date, and Datetime data types only. • `gte`—Specifies a greater than or equal to criteria. Available from API version 63.0 and later for Number, Date, and Datetime data types only. • `lte`—Specifies a less than or equal to criteria. Available from API version 63.0 and later for Number, Date, and Datetime data types only. If multiple criteria are specified, then the resultant criteria are combined by using the `and` operator.	Optional	60.0
`includeCatalogDetails`	Boolean	Indicates whether to include catalog details in the response (`true`) or not (`false`).	Optional	61.0

Name	Type	Description	Required or Optional	Available Version
`limit`	Integer	Number of items to include in the response. The default value is 10.	Optional	60.0
`offset`	Integer	Reserved for internal use.	Optional	60.0
`orderBy`	String[]	Sort order of the results, which is either ascending or descending order. The default sort order is ascending order. The default value is `asc`.	Optional	60.0
`priceBookId`	String	ID of the price book to get the prices from. If this property isn't specified, then prices from the standard price book are fetched.	Optional	60.0
`pricingProcedure`	String	API name of the custom pricing procedure that's used for the pricing process. If this property isn't specified, then the default pricing procedure is executed.	Optional	60.0
`productClassificationId`	String	ID of the product classification.	Optional	60.0
`qualificationProcedure`	String	API name of the custom qualification procedure that's used for the product qualification process. If this property isn't specified, then the default qualification procedure is executed.	Optional	60.0
`query`	Map<String, Object>>	Query to search the products.	Required	60.0
`relatedObjectFilter`	Related Object Filter Input[]	Filter records based on supported criteria for related objects. The supported object is `ProductSpecificationRecType`. The supported property is `IsCommerical`. The supported operator is `eq`. The supported values are `true` and `false`.	Optional	60.0
`searchTerm`	String	String used to get products with the product name containing the search term. See Search Considerations When Using Indexed Data.	Optional	62.0
`userContext`	User Context Input	User context details. For example, account ID or contact ID.	Optional	60.0

QOC Qualification

Input representation of the qualification request.

JSON example

```
{
  "productIds": [
    "01txx0000006i7PAAQ",
    "01txx0000006i7QAAQ",
    "01txx0000006i7IAAQ"
  ],
  "userContext": {
    "accountId": "001xx000003GZHgAAO"
  }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
additionalContextData	Context Data Input[]	Additional nodes that are added to the custom or default context definition. The maximum number of supported nodes is 10.	Optional	60.0
catalogId	String	ID of the catalog.	Optional	60.0
categoryId	String	ID of the category.	Optional	60.0
contextDefinition	String	API name of the custom context definition that's sent for context creation. If this property isn't specified, the default context definition is used.	Optional	60.0
contextMapping	String	Default context mapping of the context definition. If a context mapping is specified, then the API checks whether the mapping belongs to the specified context definition to process the details for hydration.	Optional	60.0
correlationId	String	Unique identifier of the request.	Optional	60.0
productIds	String[]	List of product IDs for qualification check.	Required	60.0
qualificationProcedure	String	API name of the custom qualification procedure that's sent for processing. If this property isn't specified, the default qualification procedure is executed.	Optional	60.0
userContext	User Context Input	User context details. For example, account ID or contact ID.	Optional	60.0

Related Object Filter Input

Input representation of the request to filter records of a related object.

JSON example

```
"relatedObjectFilters":
 [
    {
      "objectName": "ProductSpecificationRecType",
      "criteria":
 [{
      "property": "IsCommercial",
      "operator": "eq",
      "value": true
   }]
  }
  ]
```

Properties

Name	Type	Description	Required or Optional	Available Version
`criteria`	Filter Criteria Input[]	Criteria to filter the related objects.	Required if the `relatedObjectFilters` property is specified.	60.0
`objectName`	String	Name of the object that's related to the main object.	Required if the `relatedObjectFilters` property is specified.	60.0

User Context Input

Input representation of the details with the user context.

JSON example

```
    "userContext": {
        "accountId": "001xx0000000001AAA",
        "contactId": "003xx00000000D7AAI",
        "contextId": "e055bb18-d4e8-41c3-881e-0132b9561708"
    }
```

Properties

Name	Type	Description	Required or Optional	Available Version
`accountId`	String	ID of the account in a user context.	Optional	60.0
`contactId`	String	ID of the contact in a user context.	Optional	60.0
`contextId`	String	ID of the context that represents the created session.	Optional	60.0

Response Bodies

Learn more about the available Product Discovery API response bodies.

API Status
Output representation of the API status.

CPQ Base Bulk Details
Output representation of the response that contains the details of multiple products.

CPQ Base Details
Output representation of the catalog, category, or product details based on the request.

CPQ Base List
Output representation of the list of catalogs, categories, or products based on the request.

CPQ Message
Output representation of the API messages.

Facet Value
Output representation of the facet values found in the search result.

Guided Selection
Output representation of the details of a guided selection.

Guided Selection Search Term
Output representation of the search term details for a guided selection.

Search Products Facet
Output representation of the details of the faceted search.

User Context
Output representation of the user context details.

API Status

Output representation of the API status.

Property Name	Type	Description	Filter Group and Version	Available Version
`messages`	CPQ Message[]	Status messages of the API execution.	Small, 60.0	60.0
`statusCode`	String	Status code of the API execution.	Small, 60.0	60.0
`statusMessage`	String	Display label for the API status.	Small, 60.0	60.0

CPQ Base Bulk Details

Output representation of the response that contains the details of multiple products.

JSON example

```
{
  "apiStatus": {
    "messages": [],
```

```
    "statusCode": "FetchedDetailsSuccessfully"
  },
  "contextId": "c68c7c7e85f3ea5b0e7bcfefc0f2dba9bbd24bfe2f4240ca589af50e473e2242",
  "correlationId": "de9a674c-1807-438c-ac78-2c96f4655325",
  "result": [
    {
      "additionalFields": [],
      "attributeCategories": [],
      "attributes": [],
      "catalogs": [],
      "childProducts": [],
      "id": "01txx0000006ivJAAQ",
      "isActive": true,
      "isAssetizable": true,
      "isSoldOnlyWithOtherProds": false,
      "name": "iPhone12",
      "nodeType": "simpleProduct",
      "prices": [
        {
          "currencyIsoCode": "USD",
          "isDefault": false,
          "isSelected": true,
          "price": 100,
          "priceBookEntryId": "01uxx0000008zUkAAI",
          "priceBookId": "01sxx0000005qxxAAA",
          "pricingModel": {
            "id": "0jPxx000000009hEAA",
            "name": "OneTime",
            "pricingModelType": "OneTime"
          }
        },
        {
          "currencyIsoCode": "USD",
          "isDefault": true,
          "isSelected": false,
          "price": 15,
          "priceBookEntryId": "01uxx0000008zUmAAI",
          "priceBookId": "01sxx0000005qxxAAA",
          "pricingModel": {
            "frequency": "Months",
            "id": "0jPxx000000009iEAA",
            "name": "Monthly",
            "occurrence": 1,
            "pricingModelType": "TermDefined"
          }
        }
      ],
      "productClassification": {},
      "productCode": "iPhone12",
      "productComponentGroups": [],
      "productSellingModelOptions": [
        {
          "id": "0iOxx00000000EfEAI",
          "productId": "01txx0000006ivJAAQ",
```

```
        "productSellingModel": {
          "id": "0jPxx000000009iEAA",
          "name": "Monthly",
          "pricingTerm": 1,
          "pricingTermUnit": "Months",
          "sellingModelType": "TermDefined",
          "status": "Active"
        },
        "productSellingModelId": "0jPxx000000009iEAA"
      },
      {
        "id": "0iOxx00000000EgEAI",
        "productId": "01txx0000006ivJAAQ",
        "productSellingModel": {
          "id": "0jPxx000000009hEAA",
          "name": "OneTime",
          "sellingModelType": "OneTime",
          "status": "Active"
        },
        "productSellingModelId": "0jPxx000000009hEAA"
      }
    ],
    "productSpecificationType": {
      "name": "ProdSpecRecType1",
      "productSpecificationRecordType": {}
    },
    "qualificationContext": {
      "isQualified": true
    }
  },
  {
    "additionalFields": [],
    "attributeCategories": [],
    "attributes": [],
    "catalogs": [],
    "childProducts": [],
    "id": "01txx0000006ivLAAQ",
    "isActive": true,
    "isAssetizable": true,
    "isSoldOnlyWithOtherProds": false,
    "name": "iPhone13",
    "nodeType": "simpleProduct",
    "prices": [
      {
        "currencyIsoCode": "USD",
        "isDefault": true,
        "isSelected": false,
        "price": 1520,
        "priceBookEntryId": "01uxx0000008zUpAAI",
        "priceBookId": "01sxx0000005qxxAAA",
        "pricingModel": {
          "id": "0jPxx000000009hEAA",
          "name": "OneTime",
          "pricingModelType": "OneTime"
```

```
        }
      },
      {
        "currencyIsoCode": "USD",
        "isDefault": false,
        "isSelected": false,
        "price": 152,
        "priceBookEntryId": "01uxx0000008zUqAAI",
        "priceBookId": "01sxx0000005qxxAAA",
        "pricingModel": {
          "frequency": "Months",
          "id": "0jPxx000000009iEAA",
          "name": "Monthly",
          "occurrence": 1,
          "pricingModelType": "TermDefined"
        }
      }
    ],
    "productClassification": {},
    "productCode": "iPhone13",
    "productComponentGroups": [],
    "productSellingModelOptions": [
      {
        "id": "0iOxx00000000EbEAI",
        "productId": "01txx0000006ivLAAQ",
        "productSellingModel": {
          "id": "0jPxx000000009iEAA",
          "name": "Monthly",
          "pricingTerm": 1,
          "pricingTermUnit": "Months",
          "sellingModelType": "TermDefined",
          "status": "Active"
        },
        "productSellingModelId": "0jPxx000000009iEAA"
      },
      {
        "id": "0iOxx00000000EeEAI",
        "productId": "01txx0000006ivLAAQ",
        "productSellingModel": {
          "id": "0jPxx000000009hEAA",
          "name": "OneTime",
          "sellingModelType": "OneTime",
          "status": "Active"
        },
        "productSellingModelId": "0jPxx000000009hEAA"
      }
    ],
    "productSpecificationType": {
      "name": "ProdSpecRecType1",
      "productSpecificationRecordType": {}
    },
    "qualificationContext": {
      "isQualified": true
    }
```

```
    }
  ]
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
apiStatus	API Status[]	Status of the API request.	Small, 61.0	61.0
contextId	String	ID of the context.	Small, 61.0	61.0
correlationId	String	Unique identifier of the request.	Small, 61.0	61.0
result	Any response body	Result that contains the details of products.	Small, 61.0	61.0
userContext	User Context[]	User context details. For example, account ID or contact ID.	Small, 61.0	61.0

CPQ Base Details

Output representation of the catalog, category, or product details based on the request.

JSON example

This example shows the sample catalog details.

```
{
  "apiStatus": {
    "messages": [

    ],
    "statusCode": "FetchedDetailsSuccessfully"
  },
  "correlationId": "32595ed6-1922-41f7-9c2c-373c677a7d62",
  "result": {
    "catalogCode": "Mobiles",
    "catalogType": "Sales",
    "description": "Catalog for mobile phones",
    "effectiveEndDate": "2028-04-01T19:00Z",
    "effectiveStartDate": "2024-04-01T19:00Z",
    "id": "0ZSxx0000000001GAA",
    "name": "Mobiles",
    "numberOfCategories": 3
  }
}
```

This example shows the sample category details.

```
{
  "apiStatus": {
    "messages": [

    ],
    "statusCode": "FetchedDetailsSuccessfully"
  },
  "correlationId": "f9fb90de-36aa-44a1-9961-a9ef5fc0cad8",
```

```
  "result": {
    "catalogId": "0ZSxx0000000001GAA",
    "childCategories": [

    ],
    "description": "Category for Samsung phones",
    "id": "0ZGxx000000004rGAA",
    "name": "Samsung",
    "isNavigational": true,
    "sortOrder": 2
  }
}
```


This example shows the sample product details.


```
{
  "apiStatus": {
    "messages": [

    ],
    "statusCode": "FetchedDetailsSuccessfully"
  },
  "correlationId": "822a8941-7412-4883-bb80-e488f908471e",
  "result": {
    "additionalFields": {
        "CustomField1__c": "TextValue",
        "CustomField2__c": "10",
        "StandardField1": "false"
    },
    "attributeCategories": [

    ],
    "attributes": [

    ],
    "catalogs": [

    ],
    "childProducts": [

    ],
    "id": "01txx0000006i2WAAQ",
    "isActive": true,
    "isAssetizable": true,
    "isSoldOnlyWithOtherProds": false,
    "name": "iPhone15",
    "nodeType": "simpleProduct",
    "prices": [

    ],
    "productClassification": {

    },
    "productCode": "iPhone15",
    "productComponentGroups": [
```

```
    ],
    "productSellingModelOptions": [
      {
        "id": "0iOxx0000000003EAA",
        "productId": "01txx0000006i2WAAQ",
        "productSellingModel": {
          "id": "0jPxx0000000001EAA",
          "name": "OneTime",
          "sellingModelType": "OneTime",
          "status": "Active"
        },
        "productSellingModelId": "0jPxx0000000001EAA"
      },
      {
        "id": "0iOxx0000000004EAA",
        "productId": "01txx0000006i2WAAQ",
        "productSellingModel": {
          "id": "0jPxx0000000002EAA",
          "name": "Monthly",
          "pricingTerm": 1,
          "pricingTermUnit": "Months",
          "sellingModelType": "TermDefined",
          "status": "Active"
        },
        "productSellingModelId": "0jPxx0000000002EAA"
      }
    ],
    "productSpecificationType": {
      "name": "ProdSpecRecType1",
      "productSpecificationRecordType": {

      }
    }
  }
}
```


Property Name	Type	Description	Filter Group and Version	Available Version
apiStatus	API Status	Status of the API request.	Small, 60.0	60.0
contextId	String	ID of the context.	Small, 60.0	60.0
correlationId	String	Unique identifier of the request.	Small, 60.0	60.0
result	Any response body	Result that contains the details of catalogs, categories, or products as per the requested resource.	Small, 60.0	60.0
userContext	User Context	User context details. For example, account ID or contact ID.	Small, 60.0	60.0

CPQ Base List

Output representation of the list of catalogs, categories, or products based on the request.

JSON example

This example shows a sample catalog list.

```
{
  "apiStatus": {
    "messages": [

    ],
    "statusCode": "FetchedDetailsSuccessfully"
  },
  "correlationId": "9f417514-9587-4063-9e48-18a2cf2477c0",
  "limit": 10,
  "offSet": 0,
  "query": {

  },
  "result": [
    {
      "catalogCode": "Mobiles",
      "catalogType": "Sales",
      "description": "Catalog for mobile phones",
      "effectiveEndDate": "2028-04-01T19:00Z",
      "effectiveStartDate": "2024-04-01T19:00Z",
      "id": "0ZSxx0000000001GAA",
      "name": "Mobiles",
      "numberOfCategories": 3
    }
  ],
  "total": 1
}
```

This example shows a sample category list.

```
{
  "apiStatus": {
    "messages": [

    ],
    "statusCode": "FetchedDetailsSuccessfully"
  },
  "correlationId": "3f2a8f45-e7d2-42ec-bc4c-b981d750e912",
  "query": {

  },
  "result": [
    {
      "catalogId": "0ZSxx0000000001GAA",
      "childCategories": [

      ],
      "description": "Category for Apple phones and iPads",
      "id": "0ZGxx0000000001GAA",
```

```
      "name": "Apple",
      "sortOrder": 1,
      "isNavigational": true
    },
    {
      "catalogId": "0ZSxx0000000001GAA",
      "childCategories": [

      ],
      "description": "Category for Samsung phones",
      "id": "0ZGxx000000004rGAA",
      "name": "Samsung",
      "sortOrder": 2,
      "isNavigational": true
    },
    {
      "catalogId": "0ZSxx0000000001GAA",
      "childCategories": [

      ],
      "description": "Category for Android phones",
      "id": "0ZGxx000000006TGAQ",
      "name": "Android",
      "isNavigational": true
    }
  ],
  "total": 3
}
```


This example shows a sample product list.


```
{
  "apiStatus": {
    "messages": [],
    "statusCode": "FetchedDetailsSuccessfully"
  },
  "contextId": "f36f8e73f1fc338cc4e93c61613cba07a6a0129941d97e5dd6e52a2885776ce4",
  "correlationId": "eeaa1db2-f371-4227-a886-c77e2f66ce1d",
  "cursor": "MTAwMDAwMDAwNg==",
  "query": {},
  "result": [
    {
      "additionalFields": {
        "DecompositionScope": "OrderLineItem",
        "ProductCode": "LPB001",
        "CanRamp": false
      },
      "attributeCategories": [],
      "catalogs": [
        {
          "customFields": {},
          "id": "0ZSDU0000002Og74AE",
          "name": "Service Catalog",
          "numberOfCategories": 5
        }
```

```
      ],
      "categories": [
        {
          "catalogId": "0ZSDU0000002Og74AE",
          "childCategories": [],
          "customFields": {},
          "hasSubCategories": false,
          "id": "0ZGDU0000002P0H4AU",
          "name": "Cloud Services",
          "qualificationContext": {
            "isQualified": true
          }
        }
      ],
      "childProducts": [],
      "configureDuringSale": "Allowed",
      "description": "The laptop pro bundle includes a Laptop, mouse, warranty for 2
years, premium support and printer bundle",
      "displayUrl":
"https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcSCrGjPR1fvJqg4yP3RMyqjI0H9eL6tk1fvzw&amp;usqp=CAU",

      "id": "01tDU000000ExkZYAS",
      "isActive": true,
      "isAssetizable": true,
      "isSoldOnlyWithOtherProds": false,
      "name": "Laptop Pro Bundle",
      "nodeType": "bundleProduct",
      "prices": [],
      "productClassification": {},
      "productCode": "LPB001",
      "productComponentGroups": [],
      "productSellingModelOptions": [
        {
          "id": "0iODU0000002TBN2A2",
          "productId": "01tDU000000ExkZYAS",
          "productSellingModel": {
            "id": "0jPDU0000002OTv2AM",
            "name": "One Time",
            "sellingModelType": "OneTime",
            "status": "Active"
          },
          "productSellingModelId": "0jPDU0000002OTv2AM"
        }
      ],
      "productSpecificationType": {
        "name": "Commercial",
        "productSpecificationRecordType": {}
      },
      "productType": "Bundle",
      "qualificationContext": {
        "isQualified": true
      }
    }
```

```
  ]
}
```

This example shows a sample of the list of products retrieved based on the `Laptop` search term.


```
{
  "apiStatus": {
    "messages": [],
    "statusCode": "FetchedDetailsSuccessfully"
  },
  "correlationId": "d9d8f898-19f5-464a-ba2b-6a070783f6c4",
  "cursor": "MTAwMDAwMDAwMw==",
  "facets": [
    {
      "attributeType": "ProductStandard",
      "displayName": "Product Type",
      "displayRank": 2,
      "nameOrId": "Type",
      "values": [
        {
          "displayName": "Bundle",
          "nameOrId": "Bundle"
        }
      ]
    },
    {
      "attributeType": "ProductDynamicAttribute",
      "displayName": "Display",
      "displayRank": 3,
      "nameOrId": "0tjDU0000003K5BYAU",
      "values": [
        {
          "displayName": "1080p Built-in Display",
          "nameOrId": "1080p Built-in Display"
        },
        {
          "displayName": "2k Built-in Display",
          "nameOrId": "2k Built-in Display"
        },
        {
          "displayName": "4k Built-in Display",
          "nameOrId": "4k Built-in Display"
        }
      ]
    }
  ],
  "limit": 10,
  "query": {},
  "result": [
    {
      "additionalFields": {},
      "attributeCategories": [],
      "catalogs": [],
      "categories": [
        {
```

```
          "catalogId": "0ZSDU0000002Og64AE",
          "childCategories": [],
          "customFields": {},
          "hasSubCategories": false,
          "id": "0ZGDU0000002P0A4AU",
          "name": "Laptops",
          "qualificationContext": {
            "isQualified": true
          }
        }
      ],
      "configureDuringSale": "Allowed",
      "description": "Battery- or AC-powered personal computer (PC) smaller than a
briefcase",
      "displayUrl":
"https://[illegible]",

      "id": "01tDU000000ExkWYAS",
      "isActive": true,
      "isAssetizable": true,
      "isSoldOnlyWithOtherProds": false,
      "name": "Laptop",
      "nodeType": "simpleProduct",
      "prices": [],
      "productClassification": {
        "id": "11BDU0000002TCD2A2"
      },
      "productCode": "LP001",
      "productComponentGroups": [],
      "productSellingModelOptions": [
        {
          "id": "0iODU0000002TBF2A2",
          "productId": "01tDU000000ExkWYAS",
          "productSellingModel": {
            "id": "0jPDU0000002OTv2AM",
            "name": "One Time",
            "sellingModelType": "OneTime",
            "status": "Active"
          },
          "productSellingModelId": "0jPDU0000002OTv2AM"
        }
      ],
      "productSpecificationType": {
        "name": "Commercial",
        "productSpecificationRecordType": {}
      },
      "qualificationContext": {
        "isQualified": true
      }
    },
    {
      "additionalFields": {},
      "attributeCategories": [],
      "catalogs": [],
```

```
      "categories": [
        {
          "catalogId": "0ZSDU0000002Og64AE",
          "childCategories": [],
          "customFields": {},
          "hasSubCategories": false,
          "id": "0ZGDU0000002P0A4AU",
          "name": "Laptops",
          "qualificationContext": {
            "isQualified": true
          }
        }
      ],
      "configureDuringSale": "Allowed",
      "description": "The laptop basic bundle includes a Laptop, mouse, and warranty
for 1 year.",
      "displayUrl":
"https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcTbf49JG4zZogCmZMJuXU38qOkR9X36MN4bSw&amp;usqp=CAU",

      "id": "01tDU000000ExkXYAS",
      "isActive": true,
      "isAssetizable": true,
      "isSoldOnlyWithOtherProds": false,
      "name": "Laptop Basic Bundle",
      "nodeType": "bundleProduct",
      "prices": [],
      "productClassification": {},
      "productCode": "LB001",
      "productComponentGroups": [],
      "productSellingModelOptions": [
        {
          "id": "0iODU0000002TBD2A2",
          "productId": "01tDU000000ExkXYAS",
          "productSellingModel": {
            "id": "0jPDU0000002OTv2AM",
            "name": "One Time",
            "sellingModelType": "OneTime",
            "status": "Active"
          },
          "productSellingModelId": "0jPDU0000002OTv2AM"
        }
      ],
      "productSpecificationType": {
        "name": "Commercial",
        "productSpecificationRecordType": {}
      },
      "productType": "Bundle",
      "qualificationContext": {
        "isQualified": true
      }
    },
    {
      "additionalFields": {},
      "attributeCategories": [],
```

```
      "catalogs": [],
      "categories": [
        {
          "catalogId": "0ZSDU0000002Og64AE",
          "childCategories": [],
          "customFields": {},
          "hasSubCategories": false,
          "id": "0ZGDU0000002P0A4AU",
          "name": "Laptops",
          "qualificationContext": {
            "isQualified": true
          }
        }
      ],
      "configureDuringSale": "Allowed",
      "description": "The laptop pro bundle includes a Laptop, mouse, warranty for 2
years, premium support and printer bundle",
      "displayUrl":
"https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcSCrGjPR1fvJqg4yP3RMyqjI0H9eL6tk1fvzw&amp;usqp=CAU",

      "id": "01tDU000000ExkZYAS",
      "isActive": true,
      "isAssetizable": true,
      "isSoldOnlyWithOtherProds": false,
      "name": "Laptop Pro Bundle",
      "nodeType": "bundleProduct",
      "prices": [],
      "productClassification": {},
      "productCode": "LPB001",
      "productComponentGroups": [],
      "productSellingModelOptions": [
        {
          "id": "0iODU0000002TBN2A2",
          "productId": "01tDU000000ExkZYAS",
          "productSellingModel": {
            "id": "0jPDU0000002OTv2AM",
            "name": "One Time",
            "sellingModelType": "OneTime",
            "status": "Active"
          },
          "productSellingModelId": "0jPDU0000002OTv2AM"
        }
      ],
      "productSpecificationType": {
        "name": "Commercial",
        "productSpecificationRecordType": {}
      },
      "productType": "Bundle",
      "qualificationContext": {
        "isQualified": true
      }
    },
    {
      "additionalFields": {},
```

```
      "attributeCategories": [],
      "catalogs": [],
      "categories": [
        {
          "catalogId": "0ZSDU0000002Og64AE",
          "childCategories": [],
          "customFields": {},
          "hasSubCategories": false,
          "id": "0ZGDU0000002P0A4AU",
          "name": "Laptops",
          "qualificationContext": {
            "isQualified": true
          }
        }
      ],
      "configureDuringSale": "Allowed",
      "description": "Laptop, Laptop Bag, Laptop stand, Mouse, Keyboard, USB-C
Hub,External Hard Drive, Noise Cancelling Headphones, office 365",
      "displayUrl": "https://m.media-amazon.com/images/I/613Fno-NLYL._AC_SL1000_.jpg",
      "id": "01tDU000000ExlAYAS",
      "isActive": true,
      "isAssetizable": true,
      "isSoldOnlyWithOtherProds": false,
      "name": "Laptop Productivity Bundle",
      "nodeType": "bundleProduct",
      "prices": [],
      "productClassification": {},
      "productCode": "LPB001",
      "productComponentGroups": [],
      "productSellingModelOptions": [
        {
          "id": "0iODU0000002TBq2AM",
          "productId": "01tDU000000ExlAYAS",
          "productSellingModel": {
            "id": "0jPDU0000002OTv2AM",
            "name": "One Time",
            "sellingModelType": "OneTime",
            "status": "Active"
          },
          "productSellingModelId": "0jPDU0000002OTv2AM"
        }
      ],
      "productSpecificationType": {
        "name": "Commercial",
        "productSpecificationRecordType": {}
      },
      "productType": "Bundle",
      "qualificationContext": {
        "isQualified": true
      }
    }
  ],
  "total": 4
}
```

This example shows a sample of the results of a qualification procedure that's executed on a list of product IDs.

```
{
  "apiStatus": {
    "messages": [

    ],
    "statusCode": "FetchedDetailsSuccessfully"
  },
  "contextId": "e055bb18-d4e8-41c3-881e-0132b9561708",
  "correlationId": "c280c1b0-fd3f-4eac-9b08-075bdf1cbefc",
  "query": {

  },
  "result": [
    {
      "productId": "01txx0000006i7PAAQ",
      "qualificationContext": {
        "isQualified": true
      }
    },
    {
      "productId": "01txx0000006i7QAAQ",
      "qualificationContext": {
        "isQualified": true
      }
    },
    {
      "productId": "01txx0000006i7IAAQ",
      "qualificationContext": {
        "isQualified": true
      }
    }
  ]
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
apiStatus	API Status	Status of the API request.	Small, 60.0	60.0
contextId	String	ID of the context.	Small, 60.0	60.0
correlationId	String	Unique ID of the request.	Small, 60.0	60.0
cursor	String	Unique ID to represent the position of each product in the dataset.	Small, 60.0	60.0
facets	Search Products Facet	Details of the faceted search.	Small, 63.0	63.0
limit	Integer	Number of items fetched in the response.	Small, 60.0	60.0
offSet	Integer	Offset size from which the item count is fetched.	Small, 60.0	60.0

Property Name	Type	Description	Filter Group and Version	Available Version
query	Map<String, Object>>	Query that was used for the search request.	Small, 60.0	60.0
result	Any response body	Result that contains the list of catalogs, categories, or products as per the requested resource.	Small, 60.0	60.0
total	Integer	Number of fetched records.	Small, 60.0	60.0
userContext	User Context	User context details. For example, account ID or contact ID.	Small, 60.0	60.0

CPQ Message

Output representation of the API messages.

Property Name	Type	Description	Filter Group and Version	Available Version
code	String	Code specifying the type of message. Valid value is CartValidationError.	Small, 60.0	60.0
detail	String	Required details other than the message text.	Small, 60.0	60.0
message	String	Text of the API message.	Small, 60.0	60.0
severity	String	Severity of the API message. Valid values are: • Error • Info • Warning	Small, 60.0	60.0

Facet Value

Output representation of the facet values found in the search result.

JSON example

```
"values": [
  {
    "displayName": "1080p Built-in Display",
    "nameOrId": "1080p Built-in Display"
  },
  {
    "displayName": "2k Built-in Display",
    "nameOrId": "2k Built-in Display"
  },
```

```
        {
          "displayName": "4k Built-in Display",
          "nameOrId": "4k Built-in Display"
        }
      ]
```

Property Name	Type	Description	Filter Group and Version	Available Version
displayName	String	Display name of the facet value.	Small, 63.0	63.0
nameOrId	String	ID or the internal name of the facet value.	Small, 63.0	63.0

Guided Selection

Output representation of the details of a guided selection.

JSON example

```
{
  "apiStatus": {
    "messages": [],
    "statusCode": "FETCHED_DETAILS_SUCCESSFULLY"
  },
  "correlationId": "corrId",
  "cursor": "MTAwMDAwMDAwNg==",
  "searchTerms": [
    {
      "term": "IPhone"
    },
    {
      "term": "4GB"
    },
    {
      "term": "64GB"
    }
  ],
  "result": [
    {
      "additionalFields": {
        "CustomField1__c": "TextValue",
        "CustomField2__c": "10",
        "StandardField1": "false"
      },
      "description": "IPhone-13",
      "id": "01txx0000006kYwAAI",
      "name": "Sample product 1",
      "prices": [
        {
          "price": 150,
          "priceBookEntryId": "12Axx0000004DF7EAM",
          "priceBookId": "01sxx0000005puLAAQ",
          "pricingModel": {
            "frequency": "Monthly",
```

```
            "id": "12Bxx000000CiCDEA0",
            "name": "IPhone-13",
            "occurrence": 6,
            "pricingModelType": "Recurring"
          }
        },
        {
          "price": 400,
          "priceBookEntryId": "12Axx0000004DGjEAM",
          "priceBookId": "01sxx0000005puLAAQ",
          "pricingModel": {
            "id": "12Bxx000000CiCCEA0",
            "name": "IPhone-13",
            "pricingModelType": "OneTime"
          }
        }
      ],
      "qualificationContext": {
        "isQualified": true
      }
    }
  ]
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
searchTerms	Guided Selection Search Term[]	Search terms for the guided selection.	Small, 62.0	62.0

Guided Selection Search Term

Output representation of the search term details for a guided selection.

JSON example

```
  "searchTerms": [
    {
      "term": "IPhone"
    },
    {
      "term": "4GB"
    },
    {
      "term": "64GB"
    }
  ]
```

Property Name	Type	Description	Filter Group and Version	Available Version
term	String	Search term value for the guided selection.	Small, 62.0	62.0

Search Products Facet

Output representation of the details of the faceted search.

JSON example

```
"facets": [
  {
    "attributeType": "ProductStandard",
    "displayName": "Product Type",
    "displayRank": 2,
    "nameOrId": "Type",
    "values": [
      {
        "displayName": "Bundle",
        "nameOrId": "Bundle"
      }
    ]
  },
  {
    "attributeType": "ProductDynamicAttribute",
    "displayName": "Display",
    "displayRank": 3,
    "nameOrId": "0tjDU0000003K5BYAU",
    "values": [
      {
        "displayName": "1080p Built-in Display",
        "nameOrId": "1080p Built-in Display"
      },
      {
        "displayName": "2k Built-in Display",
        "nameOrId": "2k Built-in Display"
      },
      {
        "displayName": "4k Built-in Display",
        "nameOrId": "4k Built-in Display"
      }
    ]
  }
]
```

Property Name	Type	Description	Filter Group and Version	Available Version
`attributeType`	String	Search attribute type of the facet.	Small, 63.0	63.0
`displayName`	String	Display name of the facet.	Small, 63.0	63.0
`displayRank`	Integer	Display rank for the facet.	Small, 63.0	63.0
`nameOrId`	String	ID or the internal name of the facet.	Small, 63.0	63.0
`values`	Facet Value[]	Values of the facet found in the search result. Sorted by display name in alphabetical order.	Medium, 63.0	63.0

User Context

Output representation of the user context details.

Property Name	Type	Description	Filter Group and Version	Available Version
`contextId`	String	ID of the context.	Small, 60.0	60.0

Product Discovery Standard Invocable Actions

Use the standard invocable actions available with Product Discovery to find and retrieve the product details.

Find Products Action
Search for the products from a catalog, category, or subcategory by using the specified search term.

Get Products Action
Get products from the specified catalog, category, or subcategory, including product qualification and pricing details.

Get Product Details Action
Get details such as attributes, hierarchy, and cardinality for the specified product.

Find Products Action

Search for the products from a catalog, category, or subcategory by using the specified search term.

This action is available in API version 62.0 and later.

Special Access Rules

The Find Products action is available in Enterprise, Unlimited, and Developer Editions where Product Discovery is enabled.

Supported REST HTTP Methods

URI

`/services/data/v63.0/actions/standard/findProducts`

Formats

JSON, XML

HTTP Methods

POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`additionalContextData`	**Type** Apex-defined **Description** An array of Apex `runtime_industries_cpq.AdditionalContextData` records that contain the additional nodes that are used along with the context definition nodes for data hydration. The maximum number of supported nodes is 10.
`additionalFields`	**Type** Apex-defined **Description** An Apex `runtime_industries_cpq.AdditionalFields` record that contains an array of additional standard or custom fields to include in the response. The supported objects are: • `Product2` • `ProductAttributeDefinition`—If the fields defined for the `ProductAttributeDefinition` object aren't available for the `ProductClassificationAttr` object, then the API request fails.
`catalogId`	**Type** string **Description** Catalog ID that's used to find and retrieve the products.
`categoryId`	**Type** string **Description** ID of the category or subcategory to get the products for.
`contextDefinition`	**Type** string **Description** API name of the context definition used for context creation. If you don't specify a value, the context selected on the Product Discovery Settings page from Setup is used.
`contextMapping`	**Type** string

Input	Details
	Description API name of the context mapping that's used for data hydration. The value of this parameter is used only if it belongs to the specified context definition.
`correlationId`	**Type** string **Description** Currency code that's used to calculate and show prices. Only the products with the currency code matching the specified currency code are fetched.
`currencyCode`	**Type** string **Description** Currency code that's used to calculate and show prices.
`cursor`	**Type** string **Description** A unique identifier that represents the position of the product from which the next set of results are retrieved.
`enablePricing`	**Type** boolean **Description** Indicates whether the pricing procedure must run (`true`) or not (`false`). The default value is `true`. To use this parameter, you must enable the Pricing Procedure setting from Setup.
`enableQualification`	**Type** boolean **Description** Indicates whether the qualification procedure must run (`true`) or not (`false`). The default value is `true`. To use this parameter, you must enable the Qualification Procedure setting from Setup.
`filter`	**Type** Apex-defined

Input	Details
	Description A collection of Apex `runtime_industries_cpq.FilterInputRepresentation` records where each record contains a related object and the filter criteria that's applied on the object. The `filter` parameter supports only the `name` property. The supported operators are: • `eq` • `in` • `contains` If this parameter contains multiple criteria, all the criteria are applied.
`includeCatalogDetails`	**Type** boolean **Description** Indicates whether catalog details must be included in the response (`true`) or not (`false`).
`limit`	**Type** integer **Description** Maximum number of results to be returned in the response. Enter a value from 1 through 100. The default value is 10.
`orderBy`	**Type** string **Description** Comma-separated string of key-value pairs that specify how results are sorted. Each string must contain a field name and its sort order. For example, `["name:asc",”custom_field:asc”]`.
`priceBookId`	**Type** string **Description** ID of the pricebook from which you wan to retrieve the pricing details.
`productClassificationId`	**Type** string **Description** ID of the product classification that's used to filter products.

Input	Details
`pricingProcedure`	**Type** string **Description** API name of the pricing procedure to calculate product prices. If you don't specify a value, the pricing procedure selected on the Product Discovery Settings page from Setup is used.
`qualificationProcedure`	**Type** string **Description** API name of the qualification procedure to evaluate product eligibility. If you don't specify a value, the qualification procedure selected on the Product Discovery Settings page from Setup is used.
`relatedObjectFilters`	**Type** Apex-defined **Description** A collection of Apex `runtime_industries_cpq.RelatedObjectFilterInputRepresentation` records, where each record contains a related object and the filter criteria that's applied on the object.
`searchTerm`	**Type** string **Description** Required. Search term to find and retrieve products.
`userContext`	**Type** Apex-defined **Description** An Apex `ConnectApi.UserContextInputRepresentation` record that contains the user details to evaluate product eligibility and calculate prices.

Outputs

Output	Details
`apiStatus`	**Type** Apex-defined

Output	Details
	Description An Apex `runtime_industries_cpq.ApiStatusRepresentation` record that contains a status code and message.
`contextId`	**Type** string **Description** ID of the context that's created by using the specified context definition.
`correlationId`	**Type** string **Description** ID to reference a series of related actions.
`cursor`	**Type** string **Description** Unique identifier that represents the position of the next product in the dataset. It's used as an input to retrieve the next set of products.
`results`	**Type** Apex-defined **Description** An Apex `runtime_industries_cpq.SearchProductsRepresentation` record that contains the products that match the query.
`userContext`	**Type** Apex-defined **Description** An Apex `ConnectApi.UserContextRepresentation` record that includes the user details.

Example

POST

This sample request is for the Find Products action.

```
{
  "inputs": [
```

```
{
  "searchTerm": "firstproduct",
  "additionalContextData": [
    {
      "nodeName": "Contract",
      "nodeData": {
        "id": "xxxxx231",
        "name": "Contract1"
      }
    },
    {
      "nodeName": "Lead",
      "nodeData": {
        "id": "lllllll31",
        "name": "Lead1"
      }
    }
  ],
  "additionalFields": {
    "Product2": {
      "fields": [
        "CustomField1__c",
        "StandardField1"
      ]
    }
  },
  "correlationId": "9cbb9650-48c5-11ed-96d1-0afcf185843b",
  "catalogId": "0ZSxx0000000001GAA",
  "categoryId": "0ZGxx0000000004TAJ",
  "currencyCode": "USD",
  "priceBookId": "01sxx0000005puLAAQ",
  "productClassificationId": "11BRO00000000222AA",
  "limit": 10,
  "cursor": "MTAwMDAwMDAwNg==",
  "orderBy": [
    "name:asc"
  ],
  "userContext": {
    "accountId": "001xx0000000001AAA"
  },
  "enableQualification": true,
  "enablePricing": true,
  "qualificationProcedure": "QualificationProcedure",
  "pricingProcedure": "Preview",
  "contextDefinition": "TestDefinition",
  "contextMapping": "TestDefinitionNode",
  "includeCatalogDetails": false,
  "filter": {
    "criteria": [
      {
        "property": "name",
        "operator": "eq",
        "value": "Catalog_Name_1"
      }
```

```
        ]
      },
      "relatedObjectFilters": [
        {
          "objectName": "ProductSpecificationRecType",
          "criteria": [
            {
              "property": "IsCommercial",
              "operator": "eq",
              "value": true
            }
          ]
        }
      ]
    }
  ]
}
```

This is the sample response for the Find Products action.

```
{
    "apiStatus": {
      "messages": [],
      "statusCode": "FETCHED_DETAILS_SUCCESSFULLY"
    },
    "contextId": "0U3RM00000000SR0AY",
    "correlationId": "9cbb9650-48c5-11ed-96d1-0afcf185843b",
    "cursor": "MTAwMDAwMDAwNg==",
    "result": [
      {
        "additionalFields": {
          "CustomField1__c": "TextValue",
          "CustomField2__c": "10",
          "StandardField1": "false"
        },
        "description": "IPhone-13",
        "id": "01txx0000006kYwAAI",
        "name": "Sample product 1",
        "prices": [
          {
            "price": 150,
            "priceBookEntryId": "12Axx0000004DF7EAM",
            "priceBookId": "01sxx0000005puLAAQ",
            "pricingModel": {
              "frequency": "Monthly",
              "id": "12Bxx000000CiCDEA0",
              "name": "IPhone-13",
              "occurrence": 6,
              "pricingModelType": "Recurring"
            }
          },
          {
            "price": 400,
            "priceBookEntryId": "12Axx0000004DGjEAM",
            "priceBookId": "01sxx0000005puLAAQ",
```

```
            "pricingModel": {
              "id": "12Bxx000000CiCCEA0",
              "name": "IPhone-13",
              "pricingModelType": "OneTime"
            }
          }
        ],
        "qualificationContext": {
          "isQualified": true
        }
      },
      {
        "additionalFields": {
          "CustomField1__c": "TextValue",
          "CustomField2__c": "10",
          "StandardField1": "false"
        },
        "description": "Sample product 2",
        "name": "Sample product 2",
        "id": "01txx0000006kYwAAI",
        "prices": [],
        "qualificationContext": {
          "isQualified": false
        }
      },
      {
        "description": "Sample product 3",
        "name": "Sample product 3",
        "id": "01txx0000006kYwAAI",
        "prices": [],
        "qualificationContext": {
          "isQualified": true
        }
      }
    ],
    "userContext": {
      "accountId": "001xx0000000001AAA"
    }
  }
```

Usage of an Apex-Defined Data Type in a Flow

To use an Apex-defined input parameter in a flow, follow these guidelines.

Create an Apex Class

Create an Apex class defining the input and output parameters. In the flow, include the Apex-defined input parameters for which you want to add the details. In this example, we've created a class named ProductServiceAction that takes an object's API name and record ID as input, and returns the additional context data.

```
public class ProductServiceAction {
    // Define input parameters
    public class FlowInput{
        @InvocableVariable(required=false)
        public String objectApiName;
```

```
        @InvocableVariable(required=false)
        public String recordId;
    }

    // Define output parameters
    public class FlowOutput{
        @invocableVariable
        public runtime_industries_cpq.AdditionalContextData
additionalContextDataFinalOutput = new runtime_industries_cpq.AdditionalContextData();
    }

    // This method is invoked from a Flow
    @InvocableMethod(label='Process Input' description='Creates the Array of
ContextDataInput for additional Context Data')
    public static List<FlowOutput> processContextData(List<FlowInput> inputs){
        String apiName;
        String recId;
        FlowOutput output = new FlowOutput();

    // Capture input from the flow
        for(FlowInput input: inputs){
            apiName = input.objectApiName;
            recId = input.recordId;
            }
    // Populate the ContextDataInput list to store additional context data
        List<runtime_industries_cpq.ContextDataInput> listContextData = new
List<runtime_industries_cpq.ContextDataInput>();
        runtime_industries_cpq.ContextDataInput cd1 = new
runtime_industries_cpq.ContextDataInput();
        cd1.nodeName = apiName;
        cd1.nodeData = new Map<String, Object>();
        cd1.nodeData.put('id',recId);
        listContextData.add(cd1);

       output.additionalContextDataFinalOutput.additionalContextData = listContextData;

        List<FlowOutput> flowOutputs = new List<FlowOutput>();
        flowOutputs.add(output);

        List<runtime_industries_cpq.RelatedObjectFilter> relatedObjectFilterList = new
 List<runtime_industries_cpq.RelatedObjectFilter>();

        runtime_industries_cpq.RelatedObjectFilter relatedObjectFilter = new
runtime_industries_cpq.RelatedObjectFilter();

        relatedObjectFilter.objectName = 'ProductSpecificationRecType';
        List<runtime_industries_cpq.FilterCriteriaInputRepresentation> criteriaList =
new  List<runtime_industries_cpq.FilterCriteriaInputRepresentation>();
        runtime_industries_cpq.FilterCriteriaInputRepresentation criteria = new
runtime_industries_cpq.FilterCriteriaInputRepresentation();
        criteria.property = 'IsCommercial';
        criteria.operator = 'eq';
        criteria.value = 'true';
```

```
        criteriaList.add(criteria);
        relatedObjectFilter.criteria = criteriaList;

        relatedObjectFilterList.add(relatedObjectFilter);
        output.relatedObjectFilter.relatedObjectFilter  = relatedObjectFilterList;

  output.userContext.accountId = '001DU000001nx9BYAQ';

        return flowOutputs;

    }

}
```

Create a Flow with the Necessary Variables and Components

Create a flow that enables users to add a search term to find products. Add the ProductService action that you've created above by using Apex. When a flow is invoked from a record, the flow sends the record's objectApiName and recordId to the Apex class, which then generates the flow output. The flow passes the objectApiName and recordId of the record that the flow is invoked from to the Apex class to generate the flow output. See Example of How to Create a Flow for Product Discovery.

Configure the Action

Configure the action (for example, Find Products action) to add values for the Apex-defined input parameters. Use the output of the created Apex class as the input of the Apex-defined parameter in the Find Products action, which users can use to find products.

Get Products Action

Get products from the specified catalog, category, or subcategory, including product qualification and pricing details.

If a catalog, category, or subcategory isn't specified, then this action retrieves all the products from all catalogs. This action is available in API version 62.0 and later.

Special Access Rules

The Get Products action is available in Enterprise, Unlimited, and Developer Editions where Product Discovery is enabled.

Supported REST HTTP Methods

URI

`/services/data/v63.0/actions/standard/getProducts`

Formats

JSON, XML

HTTP Methods

POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`additionalContextData`	**Type** Apex-defined **Description** An array of Apex `runtime_industries_cpq.AdditionalContextData` records that contain the additional nodes that are used along with the context definition nodes for data hydration. The maximum number of supported nodes is 10.
`additionalFields`	**Type** Apex-defined **Description** An Apex `runtime_industries_cpq.AdditionalFields` record that contains an array of additional standard or custom fields to include in the response. The supported objects are: • `Product2` • `ProductAttributeDefinition`—If the fields defined for the `ProductAttributeDefinition` object aren't available for the `ProductClassificationAttr` object, then the API request fails.
`catalogId`	**Type** string **Description** Catalog ID that's used to find and retrieve the products.
`categoryId`	**Type** string **Description** ID of the category or subcategory to get the products for.
`contextDefinition`	**Type** string **Description** API name of the context definition that's used for context creation. If you don't specify a value, the context selected on the Product Discovery Settings page from Setup is used.

Input	Details
`contextMapping`	**Type** string **Description** API name of the context mapping that's used for data hydration. The value of this parameter is used only if it belongs to the specified context definition.
`correlationId`	**Type** string **Description** Unique ID that's used to reference a series of related actions.
`currencyCode`	**Type** string **Description** Currency code that's used to calculate and show prices. Only the products with the currency code matching the specified currency code are fetched.
`cursor`	**Type** string **Description** Unique identifier that represents the position of the product from which the next set of results are retrieved.
`enablePricing`	**Type** boolean **Description** Indicates whether the pricing procedure must run (`true`) or not (`false`). The default value is `true`. To use this parameter, you must enable the Pricing Procedure setting from Setup.
`enableQualification`	**Type** boolean **Description** Indicates whether the qualification procedure must run (`true`) or not (`false`). The default value is `true`. To use this parameter, you must enable the Qualification Procedure setting from Setup.

Input	Details
`filter`	**Type** Apex-defined **Description** A collection of Apex `runtime_industries_cpq.FilterInputRepresentation` records where each record contains a related object and the filter criteria that's applied on the object. The `filter` parameter supports only the `name` property. The supported operators are: • `eq` • `in` • `contains` If this parameter contains multiple criteria, all the criteria are applied.
`includeCatalogDetails`	**Type** boolean **Description** Indicates whether catalog details must be included in the response (`true`) or not (`false`).
`limit`	**Type** integer **Description** Maximum number of results to be returned in the response. Specify a value from 1 through 100. The default value is 10.
`orderBy`	**Type** string **Description** Comma-separated string of key-value pairs that specify how the results are sorted. Each string must contain a field name and its sort order. For example, `["name:asc",”custom_field:asc”]`.
`priceBookId`	**Type** string **Description** ID of the pricebook that the pricing details are retrieved from.
`pricingProcedure`	**Type** string

Input	Details
	Description API name of the pricing procedure to calculate product prices. If you don't specify a value, the pricing procedure selected on the Product Discovery Settings page from Setup is used.
`productClassificationId`	**Type** string **Description** ID of the product classification that's used to filter the products.
`qualificationProcedure`	**Type** string **Description** API name of the qualification procedure to evaluate product eligibility. If you don't specify a value, the qualification procedure selected on the Product Discovery Settings page from Setup is used.
`relatedObjectFilters`	**Type** Apex-defined **Description** A collection of Apex `runtime_industries_cpq.RelatedObjectFilterInputRepresentation` records, where each record contains a related object and the filter criteria that's applied on the object.
`userContext`	**Type** Apex-defined **Description** An Apex `ConnectApi.UserContextInputRepresentation` record that contains the user details to evaluate product eligibility and calculate prices.

Outputs

Output	Details
`apiStatus`	**Type** Apex-defined **Description** An Apex `runtime_industries_cpq.ApiStatusRepresentation` record that contains a status code and message.

Output	Details
`contextId`	**Type** string **Description** ID of the context that's created by using the specified context definition.
`correlationId`	**Type** string **Description** ID to reference a series of related actions.
`cursor`	**Type** string **Description** Unique identifier that represents the position of the product from which the next set of results are retrieved.
`results`	**Type** Apex-defined **Description** An Apex `runtime_industries_cpq.ProductListRepresentation` record that contains the retrieved products.
`userContext`	**Type** Apex-defined **Description** An Apex `ConnectApi.UserContextRepresentation` record that includes the user details.

Example

POST

This sample request is for the Get Products action.

```
{
  "inputs": [
    {
      "additionalContextData": [
        {
          "nodeName": "Contract",
          "nodeData": {
            "id": "xxxxx231",
```

```
            "name": "Contract1"
          }
        },
        {
          "nodeName": "Lead",
          "nodeData": {
            "id": "lllllll31",
            "name": "Lead1"
          }
        }
      ],
      "additionalFields": {
        "Product2": {
          "fields": [
            "CustomField1__c",
            "StandardField1"
          ]
        }
      },
      "correlationId": "9cbb9650-48c5-11ed-96d1-0afcf185843b",
      "catalogId": "0ZSxx0000000001GAA",
      "categoryId": "0ZGxx0000000004TAJ",
      "currencyCode": "USD",
      "priceBookId": "01sxx0000005puLAAQ",
      "productClassificationId": "11BRO00000000222AA",
      "limit": 10,
      "cursor": "MTAwMDAwMDAwNg==",
      "orderBy": [
        "name:asc"
      ],
      "userContext": {
        "accountId": "001xx0000000001AAA"
      },
      "enableQualification": true,
      "enablePricing": true,
      "qualificationProcedure": "QualificationProcedure",
      "pricingProcedure": "Preview",
      "contextDefinition": "TestDefinition",
      "contextMapping": "TestDefinitionNode",
      "filter": {
        "criteria": [
          {
            "property": "name",
            "operator": "eq",
            "value": "Catalog_Name_1"
          }
        ]
      },
      "relatedObjectFilters": [
        {
          "objectName": "ProductSpecificationRecType",
          "criteria": [
            {
              "property": "IsCommercial",
```

```
                "operator": "eq",
                "value": true
              }
            ]
          }
        ]
      }
    ]
}
```

This is the sample response for the Get Products action.


```
{
    "apiStatus": {
      "messages": [],
      "statusCode": "FETCHED_DETAILS_SUCCESSFULLY"
    },
    "contextId": "0U3RM00000000SR0AY",
    "correlationId": "9cbb9650-48c5-11ed-96d1-0afcf185843b",
    "cursor": "MTAwMDAwMDAwNg==",
    "result": [
      {
        "additionalFields": {
          "CustomField1__c": "TextValue",
          "CustomField2__c": "10",
          "StandardField1": "false"
        },
        "description": "IPhone-13",
        "id": "01txx0000006kYwAAI",
        "name": "Sample product 1",
        "prices": [
          {
            "price": 150,
            "priceBookEntryId": "12Axx0000004DF7EAM",
            "priceBookId": "01sxx0000005puLAAQ",
            "pricingModel": {
              "frequency": "Monthly",
              "id": "12Bxx000000CiCDEA0",
              "name": "IPhone-13",
              "occurrence": 6,
              "pricingModelType": "Recurring"
            }
          },
          {
            "price": 400,
            "priceBookEntryId": "12Axx0000004DGjEAM",
            "priceBookId": "01sxx0000005puLAAQ",
            "pricingModel": {
              "id": "12Bxx000000CiCCEA0",
              "name": "IPhone-13",
              "pricingModelType": "OneTime"
            }
          }
        ],
        "qualificationContext": {
```

```
          "isQualified": true
        }
      },
      {
        "additionalFields": {
          "CustomField1__c": "TextValue",
          "CustomField2__c": "10",
          "StandardField1": "false"
        },
        "description": "Sample product 2",
        "name": "Sample product 2",
        "id": "01txx0000006kYwAAI",
        "prices": [],
        "qualificationContext": {
          "isQualified": false
        }
      },
      {
        "description": "Sample product 3",
        "name": "Sample product 3",
        "id": "01txx0000006kYwAAI",
        "prices": [],
        "qualificationContext": {
          "isQualified": true
        }
      }
    ],
    "userContext": {
      "accountId": "001xx0000000001AAA"
    }
  }
```

Usage of an Apex-Defined Data Type in a Flow

To use an Apex-defined input parameter in a Flow, follow these guidelines.

Create an Apex class

Create an Apex class defining the input and output parameters. In the flow, include the Apex-defined input parameters for which you want to add the details. In this example, we have created a class named ProductServiceAction that takes an object's API name and record ID as input, and returns the additional context data.

```
public class ProductServiceAction {
    // Define input parameters
    public class FlowInput{
        @InvocableVariable(required=false)
        public String objectApiName;

        @InvocableVariable(required=false)
        public String recordId;
    }

    // Define output parameters
    public class FlowOutput{
        @invocableVariable
```

```
        public runtime_industries_cpq.AdditionalContextData
additionalContextDataFinalOutput = new runtime_industries_cpq.AdditionalContextData();

        @invocableVariable
        public runtime_industries_cpq.RelatedObjectFilterInputRepresentation
relatedObjectFilter = new runtime_industries_cpq.RelatedObjectFilterInputRepresentation();


        @invocableVariable
        public ConnectApi.UserContextInputRepresentation userContext = new
ConnectApi.UserContextInputRepresentation();

        }

    // This method is invoked from a Flow
    @InvocableMethod(label='Process Input' description='Creates the Array of
ContextDataInput for additional Context Data')
    public static List<FlowOutput> processContextData(List<FlowInput> inputs){
        String apiName;
        String recId;
        FlowOutput output = new FlowOutput();

    // Capture input from the flow
        for(FlowInput input: inputs){
            apiName = input.objectApiName;
            recId = input.recordId;
            }
    // Populate the ContextDataInput list to store additional context data
        List<runtime_industries_cpq.ContextDataInput> listContextData = new
List<runtime_industries_cpq.ContextDataInput>();
        runtime_industries_cpq.ContextDataInput cd1 = new
runtime_industries_cpq.ContextDataInput();
        cd1.nodeName = apiName;
        cd1.nodeData = new Map<String, Object>();
        cd1.nodeData.put('id',recId);
        listContextData.add(cd1);

       output.additionalContextDataFinalOutput.additionalContextData = listContextData;

        List<FlowOutput> flowOutputs = new List<FlowOutput>();
        flowOutputs.add(output);

        List<runtime_industries_cpq.RelatedObjectFilter> relatedObjectFilterList = new
 List<runtime_industries_cpq.RelatedObjectFilter>();

        runtime_industries_cpq.RelatedObjectFilter relatedObjectFilter = new
runtime_industries_cpq.RelatedObjectFilter();

        relatedObjectFilter.objectName = 'ProductSpecificationRecType';
        List<runtime_industries_cpq.FilterCriteriaInputRepresentation> criteriaList =
new  List<runtime_industries_cpq.FilterCriteriaInputRepresentation>();
        runtime_industries_cpq.FilterCriteriaInputRepresentation criteria = new
runtime_industries_cpq.FilterCriteriaInputRepresentation();
        criteria.property = 'IsCommercial';
```

```
        criteria.operator = 'eq';
        criteria.value = 'true';
        criteriaList.add(criteria);
        relatedObjectFilter.criteria = criteriaList;

       relatedObjectFilterList.add(relatedObjectFilter);

        output.relatedObjectFilter.relatedObjectFilter  = relatedObjectFilterList;

  output.userContext.accountId = '001Hs00003r17HNIAY';

        return flowOutputs;

    }

}
```

Create a Flow with the necessary variables and components

Create a flow that enables users to add a search term to find products. Add the ProductService action that you've created above by using Apex. When a flow is invoked from a record, the flow sends the record's objectApiName and recordId to the Apex class, which then generates the flow output. The flow passes the objectApiName and recordId of the record that the flow is invoked from to the Apex class to generate the flow output. See Example of How to Create a Flow for Product Discovery.

Configure the action

Configure the action (for example, Get Products action) to add values for the Apex-defined input parameters. Use the output of the created Apex class as the input of the Apex-defined parameter in the Get Products action, which users can use to get products.

Get Product Details Action

Get details such as attributes, hierarchy, and cardinality for the specified product.

This action is available in API version 62.0 and later.

Special Access Rules

The Get Product Details action is available in Enterprise, Unlimited, and Developer Editions where Product Discovery is enabled.

Supported REST HTTP Methods

URI

`/services/data/v63.0/actions/standard/getProductDetails`

Formats

JSON, XML

HTTP Methods

POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`additionalContextData`	**Type** Apex-defined **Description** An array of Apex `runtime_industries_cpq.AdditionalContextData` records that contain the additional nodes that are used along with the context definition nodes for data hydration. The maximum number of supported nodes is 10.
`additionalFields`	**Type** Apex-defined **Description** An Apex `runtime_industries_cpq.AdditionalFields` record that contains an array of additional standard or custom fields to include in the response. The supported objects are: • `Product2` • `ProductAttributeDefinition`—If the fields defined for the `ProductAttributeDefinition` object aren't available for the `ProductClassificationAttr` object, then the API request fails.
`catalogId`	**Type** string **Description** Catalog ID that's used to find and retrieve the products.
`contextDefinition`	**Type** string **Description** API name of the context definition for context creation. If you don't provide a value, the context selected on the Product Discovery Settings page from Setup is used.
`contextMapping`	**Type** string **Description** API name of the context mapping for data hydration. The value of this parameter is used only if it belongs to the specified context definition.

Input	Details
`correlationId`	**Type** string **Description** Currency code that's used to calculate and show prices. Only the products with the currency code matching the specified currency code are fetched.
`currencyCode`	**Type** string **Description** Currency code that's used to calculate and show prices.
`enablePricing`	**Type** boolean **Description** Indicates whether the pricing procedure must run (`true`) or not (`false`). The default value is `true`. To use this parameter, you must enable the Pricing Procedure setting from Setup.
`enableQualification`	**Type** boolean **Description** Indicates whether the qualification procedure must run (`true`) or not (`false`). The default value is `true`. To use this parameter, you must enable the Qualification Procedure setting from Setup.
`priceBookId`	**Type** string **Description** ID of the pricebook from which you want to retrieve the pricing details.
`pricingProcedure`	**Type** string **Description** API name of the pricing procedure to calculate product prices. If you don't specify a value, the pricing procedure selected on the Product Discovery Settings page from Setup is used.

Input	Details
`productId`	**Type** string **Description** Required. ID of the product to get the details for.
`productSellingModelId`	**Type** string **Description** ID of the product selling model.
`qualificationProcedure`	**Type** string **Description** API name of the qualification procedure to evaluate product eligibility. If you don't specify a value, the qualification procedure selected on the Product Discovery Settings page from Setup is used.
`userContext`	**Type** Apex-defined **Description** An Apex `ConnectApi.UserContextInputRepresentation` record that contains the user details to evaluate product eligibility and calculate prices.

Outputs

Output	Details
`apiStatus`	**Type** Apex-defined **Description** An Apex `runtime_industries_cpq.ApiStatusRepresentation` record that contains a status code and message.
`contextId`	**Type** string **Description** ID of the context that's created by using the specified context definition.

Output	Details
`correlationId`	**Type** string **Description** ID to reference a series of related actions.
`results`	**Type** Apex-defined **Description** An Apex `runtime_industries_cpq.ProductDetailsRepresentation` record that contains the product details.
`userContext`	**Type** Apex-defined **Description** An Apex `ConnectApi.UserContextRepresentation` record that contains the user details.

Example

POST

This sample request is for the Get Product Details action.

```
{
  "inputs": [{
    "additionalContextData": [
      {
        "nodeName": "Contract",
        "nodeData": {
          "id": "xxxxx231",
          "name": "Contract1"
        }
      },
      {
        "nodeName": "Lead",
        "nodeData": {
          "id": "lllllll31",
          "name": "Lead1"
        }
      }
    ],
    "additionalFields": {
      "Product2": {
        "fields": [
          "CustomField1__c",
          "StandardField1"
```

```
        ]
      }
    },
    "correlationId": "9cbb9650-48c5-11ed-96d1-0afcf185843b",
    "catalogId": "0ZSxx0000000001GAA",
    "currencyCode": "USD",
    "priceBookId": "01sxx0000005puLAAQ",
    "productId": "01txx0000006j08AAA",
    "productSellingModelId": "0jPxx000000004rEAA",
    "userContext": {
      "accountId": "001xx0000000001AAA"
    },
    "enableQualification": true,
    "enablePricing": true,
    "qualificationProcedure": "QualificationProcedure",
    "pricingProcedure": "Preview",
    "contextDefinition": "TestDefinition",
    "contextMapping": "TestDefinitionNode"
  }]
}
```


This is the sample response for the Get Product Details action.


```
{
    "apiStatus": {
      "messages": [],
      "statusCode": "FETCHED_DETAILS_SUCCESSFULLY"
    },
    "contextId": "0U3RM00000000SR0AY",
    "correlationId": "9cbb9650-48c5-11ed-96d1-0afcf185843b",
    "result": [
      {
        "additionalFields": {
          "CustomField1__c": "TextValue",
          "CustomField2__c": "10",
          "StandardField1": "false"
        },
        "description": "IPhone-13",
        "id": "01txx0000006kYwAAI",
        "name": "Sample product 1",
        "prices": [
          {
            "price": 150,
            "priceBookEntryId": "12Axx0000004DF7EAM",
            "priceBookId": "01sxx0000005puLAAQ",
            "pricingModel": {
              "frequency": "Monthly",
              "id": "12Bxx000000CiCDEA0",
              "name": "IPhone-13",
              "occurrence": 6,
              "pricingModelType": "Recurring"
            }
          },
          {
            "price": 400,
```

```
            "priceBookEntryId": "12Axx0000004DGjEAM",
            "priceBookId": "01sxx0000005puLAAQ",
            "pricingModel": {
              "id": "12Bxx000000CiCCEA0",
              "name": "IPhone-13",
              "pricingModelType": "OneTime"
            }
          }
        ],
        "qualificationContext": {
          "isQualified": true
        }
      },
      {
        "additionalFields": {
          "CustomField1__c": "TextValue",
          "CustomField2__c": "10",
          "StandardField1": "false"
        },
        "description": "Sample product 2",
        "name": "Sample product 2",
        "id": "01txx0000006kYwAAI",
        "prices": [],
        "qualificationContext": {
          "isQualified": false
        }
      },
      {
        "description": "Sample product 3",
        "name": "Sample product 3",
        "id": "01txx0000006kYwAAI",
        "prices": [],
        "qualificationContext": {
          "isQualified": true
        }
      }
    ],
    "userContext": {
      "accountId": "001xx0000000001AAA"
    }
  }
```


Usage of an Apex-Defined Data Type in a Flow

To use an Apex-defined input parameter in a flow, follow these guidelines.

Create an Apex Class

Create an Apex class defining the input and output parameters. In the flow, include the Apex-defined input parameters for which you want to add the details. In this example, we have created a class named ProductServiceAction that takes an object's API name and record ID as input, and returns the additional context data.

```
public class ProductServiceAction {
    // Define input parameters
    public class FlowInput{
```

```
        @InvocableVariable(required=false)
        public String objectApiName;

        @InvocableVariable(required=false)
        public String recordId;
    }

    // Define output parameters
    public class FlowOutput{
        @invocableVariable
        public runtime_industries_cpq.AdditionalContextData
additionalContextDataFinalOutput = new runtime_industries_cpq.AdditionalContextData();

    }

    // This method is invoked from a Flow
    @InvocableMethod(label='Process Input' description='Creates the Array of
ContextDataInput for additional Context Data')
    public static List<FlowOutput> processContextData(List<FlowInput> inputs){
        String apiName;
        String recId;
        FlowOutput output = new FlowOutput();

    // Capture input from the flow
        for(FlowInput input: inputs){
            apiName = input.objectApiName;
            recId = input.recordId;
            }
    // Populate the ContextDataInput list to store additional context data
        List<runtime_industries_cpq.ContextDataInput> listContextData = new
List<runtime_industries_cpq.ContextDataInput>();
        runtime_industries_cpq.ContextDataInput cd1 = new
runtime_industries_cpq.ContextDataInput();
        cd1.nodeName = apiName;
        cd1.nodeData = new Map<String, Object>();
        cd1.nodeData.put('id',recId);
        listContextData.add(cd1);

       output.additionalContextDataFinalOutput.additionalContextData = listContextData;

        List<FlowOutput> flowOutputs = new List<FlowOutput>();
        flowOutputs.add(output);

        List<runtime_industries_cpq.RelatedObjectFilter> relatedObjectFilterList = new
 List<runtime_industries_cpq.RelatedObjectFilter>();

        runtime_industries_cpq.RelatedObjectFilter relatedObjectFilter = new
runtime_industries_cpq.RelatedObjectFilter();

        relatedObjectFilter.objectName = 'ProductSpecificationRecType';
        List<runtime_industries_cpq.FilterCriteriaInputRepresentation> criteriaList =
new  List<runtime_industries_cpq.FilterCriteriaInputRepresentation>();
        runtime_industries_cpq.FilterCriteriaInputRepresentation criteria = new
runtime_industries_cpq.FilterCriteriaInputRepresentation();
```

```
        criteria.property = 'IsCommercial';
        criteria.operator = 'eq';
        criteria.value = 'true';
        criteriaList.add(criteria);
        relatedObjectFilter.criteria = criteriaList;

        relatedObjectFilterList.add(relatedObjectFilter);
        output.relatedObjectFilter.relatedObjectFilter  = relatedObjectFilterList;

  output.userContext.accountId = '001DU000001nx9BYAQ';

        return flowOutputs;

    }

}
```

Create a Flow with the Necessary Variables and Components

Create a flow that enables users to add a search term to find products. Add the ProductService action that you've created above by using Apex. When a flow is invoked from a record, the flow sends the record's objectApiName and recordId to the Apex class, which then generates the flow output. The flow passes the objectApiName and recordId of the record that the flow is invoked from to the Apex class to generate the flow output. See Example of How to Create a Flow for Product Discovery.

Configure the action

Configure the action (for example, Get Product Details action) to add values for the Apex-defined input parameters. Use the output of the created Apex class as the input of the Apex-defined parameter in the Get Product Details action, which users can use to get the product details.

Product Discovery Metadata API Types

Metadata API enables you to access some types and feature settings that you can customize in the user interface.

Flow for Product Discovery

Represents the metadata associated with a flow. With Flow, you can create an application that takes users through a series of pages to query and update the records in the database. You can also run logic and provide branching capability based on user input to build dynamic applications.

ProductDiscoverySettings

Represents the settings for Product Discovery.

Flow for Product Discovery

Represents the metadata associated with a flow. With Flow, you can create an application that takes users through a series of pages to query and update the records in the database. You can also run logic and provide branching capability based on user input to build dynamic applications.

FlowActionCall

Product Discovery exposes additional actionType values for the FlowActionCall metadata type.

Field Name	Field Type	Description
actionType	InvocableActionType (enumeration of type string)	Required. The action type. Additional valid values for Product Discovery are: • findProducts—Search for the products from a catalog, category, or subcategory by using the specified search term. • GetProducts—Get products from the specified catalog, category, or subcategory, including product qualification and pricing details. • GetProductDetails—Get details such as attributes, hierarchy, and cardinality for the specified product.

ProductDiscoverySettings

Represents the settings for Product Discovery.

Parent Type and Manifest Access

This type extends the Metadata metadata type and inherits its fullName field.

In the package manifest, all the settings metadata types for the org are accessed using the "Settings" name. See Settings for more details.

File Suffix and Directory Location

ProductDiscoverySettings values are stored in the ProductDiscoverySettings.settings file in the settings folder. The .settings files are different from other named components, because there is only one settings file for each settings component.

Version

ProductDiscoverySettings components are available in API version 62.0 and later.

Fields

Field Name	Description
enableGuidedSelling	**Field Type** boolean **Description** Indicates whether guided product selection is enabled (true) or not (false). Using guided product selection, you can manage dynamic forms that assess user requirements and show relevant products. The default value is false. Available in API version 63.0 and later.
discoverProductsFlowNameOrgValue	**Field Type** string

Field Name	Description
	Description Name of the custom flow for browsing and adding products. If this field isn't specified, the Discover Products flow is used for browsing and adding products. Available in API version 63.0 and later.
`prodDiscBrowseContextDefOrgValue`	**Field Type** string **Description** Context definition that gets updated based on the user-selected options and provides summary data.
`prodDiscPricingEnabledOrgValue`	**Field Type** string **Description** Indicates whether pricing is enabled (`true`) or not (`false`). The default value is `false`.
`prodDiscProcedureOrgValue`	**Field Type** string **Description** Pricing procedure that calculates the list price.
`prodDiscQualEnabledOrgValue`	**Field Type** string **Description** Indicates whether product qualification is enabled (`true`) or not (`false`). The default value is `false`.
`prodDiscQualificationOrgValue`	**Field Type** string **Description** Qualification procedure that determines product eligibility.

Declarative Metadata Sample Definition

The following is an example of a ProductDiscoverySettings component.

```
<ProductDiscoverySettings xmlns="http://soap.sforce.com/2006/04/metadata">
    <enableGuidedSelling>true</enableGuidedSelling>

<discoverProductsFlowNameOrgValue>revenue_products__DiscoverProducts</discoverProductsFlowNameOrgValue>

    <prodDiscPricingEnabledOrgValue>true</prodDiscPricingEnabledOrgValue>
    <prodDiscQualEnabledOrgValue>true</prodDiscQualEnabledOrgValue>
```

```
    <prodDiscProcedureOrgValue>PricingProcedure</prodDiscProcedureOrgValue>
    <prodDiscQualificationOrgValue>QualificationProcedure</prodDiscQualificationOrgValue>


<prodDiscBrowseContextDefOrgValue>BrowseContextDefinition</prodDiscBrowseContextDefOrgValue>
</ProductDiscoverySettings>
```

The following is an example `package.xml` that references the previous definition.

```
<?xml version="1.0" encoding="UTF-8"?>
<Package xmlns="http://soap.sforce.com/2006/04/metadata">
    <types>
        <members>ProductDiscovery</members>
        <name>Settings</name>
    </types>
    <version>63.0</version>
</Package>
```

Wildcard Support in the Manifest File

The wildcard character `*` (asterisk) in the `package.xml` manifest file doesn't apply to metadata types for feature settings. The wildcard applies only when retrieving all settings, not for an individual setting. For details, see Settings. For information about using the manifest file, see Deploying and Retrieving Metadata with the Zip File.

CHAPTER 3 Salesforce Pricing

In this chapter ...

- Salesforce Pricing Standard Objects
- Salesforce Pricing Business APIs
- Salesforce Pricing Standard Invocable Actions
- Salesforce Pricing Metadata API Types
- Salesforce Pricing Tooling API Objects

Create customized price adjustment methods and pricing procedures. Determine the discounts to apply to your products and services by using price adjustment methods. Get visibility into the pricing calculation process by using pricing procedures.

SEE ALSO:

Salesforce Help: Permissions for Salesforce Pricing

EDITIONS

Available in: Lightning Experience

Available in: **Enterprise**, **Unlimited**, and **Developer** Editions of Revenue Cloud where Salesforce Pricing is enabled

Salesforce Pricing Standard Objects

The Salesforce Pricing data model provides objects and fields to manage pricing processes, such as product management and calculation and application of discounts.

AttributeAdjustmentCondition

Represents the condition applied to an attribute that determines the price of a product or service being sold. This object is available in API version 60.0 and later.

AttributeBasedAdjRule

Represents the attribute conditions in a rule associated with the attribute based adjustment made for a product or service being sold. This object is available in API version 60.0 and later.

AttributeBasedAdjustment

Represents the association between the product selling model and the price adjustment for product or service being sold based on its attributes. This object stores information about the attributes that define the price of the product or service, the discounts applied, along with its value for a given date range. This object is available in API version 60.0 and later.

AttributeDefinition

Represents a product, asset, or object attribute, for example, a hardware specification or software detail. This object is available in API version 60.0 and later.

BundleBasedAdjustment

Represents the association between the product selling model and the price adjustment for a product or service being sold as a bundle. This object stores information of the product or service's price, the discounts applied, along with its value for a given date range. This object is available in API version 60.0 and later.

CostBook

Represents the cost book that contains multiple cost book entries. This object is available in API version 61.0 and later.

ContractItemPrice

Represents the price of a product on the contract. This object is available in API version 61.0 and later.

CostBookEntry

Represents the total cost of a product or service that’s determined based on various factors that affect a product's price. For example, when a product is manufactured, the weight of the raw material can be a cost factor based on the amount of material required and its shipping cost. This object is available in API version 61.0 and later.

PriceAdjustmentSchedule

Represents a series of tiered discounts based on the number of items purchased. This object is available in API version 60.0 and later.

PriceAdjustmentTier

Represents a discount tier in a price adjustment schedule. This object is available in API version 60.0 and later.

PriceBook2

Represents a price book that contains the list of products that your org sells. This object is available in API version 60.0 and later.

PriceBookEntry

Represents a product entry (an association between a Pricebook2 and Product2) in a price book. This object is available in API version 60.0 and later.

PriceBookEntryDerivedPrice

Represents the price of a product that’s derived from another source such as a product or an asset. This object is available in API version 61.0 and later.

PricingAdjBatchJob

Represents the collective update of multiple records on their prices and other adjustments. This object is available in API version 62.0 and later.

PricingAdjBatchJobLog

Represents the report that contains a list of failed adjustment requests along with an error message that describes the reason for failure. This object is available in API version 62.0 and later.

PricingAPIExecution

Represents the pricing resolution for an pricing element determined using strategy name and formula. This object is available in API version 63.0 and later.

PricingProcedureResolution

Represents a selection for a pricing procedure to execute a pricing process from a list of pricing procedures available. This object is available in API version 60.0 and later.

PricingProcessExecution

Represents a record generated during the execution of a discovery or pricing procedure. Multiple procedures may be performed within a single API call, with each recorded in a Pricing API Execution record. This object is available in API version 63.0 and later.

ProductPriceHistoryLog

Stores historical pricing data based on the product's price range. This object is available in API version 62.0 and later.

ProductPriceRange

Represents the price range of a product determined by using a product selling model that's stored in the relevant price book. This object is available in API version 62.0 and later.

ProductSellingModel

Defines one method by which a product can be sold; for example, as a one-time sale, an evergreen subscription, or a termed subscription. If the product is sold on subscription, this object defines the subscription's term. A product can have multiple product selling models. This object is available in API version 60.0 and later.

ProductSellingModelDataTranslation

Represents the translated values of the data stored within the ProductSellingModel record's fields. This object is available in API version 61.0 and later.

ProductSellingModelOption

A junction object between Product Selling Model and Product2. This object is available in API version 60.0 and later.

SEE ALSO:

Object Reference for the Salesforce Platform: Overview of Salesforce Objects and Fields

SOAP API Developer Guide: Introduction to SOAP API

AttributeAdjustmentCondition

Represents the condition applied to an attribute that determines the price of a product or service being sold. This object is available in API version 60.0 and later.

Supported Calls

`create(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), update(), upsert()`

Fields

Field	Details
`AttributeBasedAdjRuleId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Specifies the attribute adjustment rule record for which the condition is to be applied. This field is a relationship field. **Relationship Name** AttributeBasedAdjRule **Relationship Type** Lookup **Refers To** AttributeBasedAdjRule
`AttributeDefinitionId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** Specifies the attribute definition record for which the condition is to be applied. This field is a relationship field. **Relationship Name** AttributeDefinition **Relationship Type** Lookup **Refers To** AttributeDefinition
`BooleanValue`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Specifies the boolean value of the operator. Possible values are: • `False` • `True`

Field	Details
DateTimeValue	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date time value of the attribute.
DateValue	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The date value of the attribute.
DoubleValue	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The double value of the attribute.
IntegerValue	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The integer value of the attribute.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last referred to this record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed a record related to this record.

Field	Details
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Name of the attribute adjustment condition.
`Operator`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update **Description** Operator used by the attribute. Possible values are: • `doesnotexistin`—Does Not Exist In • `equals`—Equals • `existsin`—Exists In • `greaterorequal`—Greater Or Equal • `greaterthan`—Greater Than • `lessorequal`—Less Or Equal • `lessthan`—Less Than • `matches`—Matches • `notequals`—Not Equals The default value is `equals`.
`ProductId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The ID of the product associated with the attribute adjustment condition. This field is a relationship field. **Relationship Name** Product **Relationship Type** Lookup **Refers To** Product2

Field	Details
`StringValue`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The string value of the attribute.
`UsageType`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** The type of record where the attribute adjustment condition is used. Possible values are: • `Pricing` • `Rating`

Associated Objects

This object has these associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

AttributeAdjustmentConditionFeed **on page 1673**
Feed tracking is available for the object.

AttributeAdjustmentConditionHistory **on page 1680**
History is available for tracked fields of the object.

AttributeBasedAdjRule

Represents the attribute conditions in a rule associated with the attribute based adjustment made for a product or service being sold. This object is available in API version 60.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`AttributeCount`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The number of attributes.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the attribute based adjustment rule was last referenced.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed a record related to this record.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** Name of the attribute based adjustment rule.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** ID of the owner of the attribute based adjustment rule. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup

Field	Details
	Refers To Group, User
`UsageType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The type of record where the attribute-based adjustment rule is used. Possible values are: • `Pricing` • `Rating`

Associated Objects

This object has these associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

AttributeBasedAdjRuleFeed on page 1673
Feed tracking is available for the object.

AttributeBasedAdjRuleHistory on page 1680
History is available for tracked fields of the object.

AttributeBasedAdjRuleShare on page 1684
Sharing is available for the object.

AttributeBasedAdjustment

Represents the association between the product selling model and the price adjustment for product or service being sold based on its attributes. This object stores information about the attributes that define the price of the product or service, the discounts applied, along with its value for a given date range. This object is available in API version 60.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`AdjustmentType`	**Type** picklist

Field	Details
	Properties Create, Filter, Group, Restricted picklist, Sort, Update **Description** The type of pricing adjustment being made. Possible values are: • `Amount` • `Override` • `Percentage`
`AdjustmentValue`	**Type** double **Properties** Create, Filter, Sort, Update **Description** The value of the adjustment being made based on the adjustment type.
`AttributeBasedAdjRuleId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The attribute based adjustment rule associated with this attribute based adjustment record. This field is a relationship field. **Relationship Name** AttributeBasedAdjRule **Relationship Type** Lookup **Refers To** AttributeBasedAdjRule
`AttributeCount`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The number of attributes.
`CurrencyIsoCode`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field	Details
	Description ISO code of the currency. Must be one of the valid alphabetic, three-letter currency ISO codes defined by the ISO 4217 standard, such as USD, GBP, or JPY. Must be unique within your organization. Possible values are: • `BHD`—Bahraini Dinar • `JPY`—Japanese Yen • `USD`—U.S. Dollar The default value is `USD`.
`EffectiveFrom`	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** The date and time when the price list entry comes into effect.
`EffectiveTo`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date and time till when the price list entry remains effective.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the attribute based adjustment was last referenced.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed a record related to this record.
`Name`	**Type** string

Field	Details
	Properties Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Name of the attribute based adjustment.
`PriceAdjustmentScheduleId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The price adjustment schedule associated with the attribute based adjustment record. This field is a relationship field. **Relationship Name** PriceAdjustmentSchedule **Relationship Type** Lookup **Refers To** PriceAdjustmentSchedule
`PricingTerm`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The number of pricing term units in the pricing term. Used with PricingTermUnit to define the length of the pricing term. For example, if PricingTermUnit is Months and this field is 1, the subscription is priced monthly.If the selling model is one-time, this field must be null.
`PricingTermUnit`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** The unit of time used to define the pricing term. Used with PricingTerm to define the length of the pricing term. For example, if this field is Months and PricingTerm is 1, the subscription is priced monthly. If the selling model is one-time, this field must be null. Possible values are: • `Annual`—Years • `Months` • `Quarterly` • `Semi-Annual`

Field	Details
`ProductId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The ID of the product associated with the product attribute set. This field is a relationship field. **Relationship Name** Product **Relationship Type** Lookup **Refers To** Product2
`ProductSellingModelId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort **Description** The ID of the ProductSellingModel record associated with this attribute based adjustment record. This field is a relationship field. **Relationship Name** ProductSellingModel **Relationship Type** Lookup **Refers To** ProductSellingModel
`SellingModelType`	**Type** picklist **Properties** Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort **Description** Indicates whether the product is sold as a one-time sale, an evergreen subscription, or a subscription with a defined term. Possible values are: • `Evergreen` • `OneTime`—One Time • `TermDefined`—Term-Defined

Field	Details
	The default value is `OneTime`.
`Weight__c`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Weight value of the attribute.
`WiFi_Standard__c`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** WiFi value of the attribute. Possible values are: • `Checkbox` • `Currency` • `Date` • `Datetime` • `Multipicklist` • `Number` • `Percent` • `Picklist` • `Text`

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

AttributeBasedAdjustmentFeed on page 1673
Feed tracking is available for the object.

AttributeBasedAdjustmentHistory on page 1680
History is available for tracked fields of the object.

AttributeDefinition

Represents a product, asset, or object attribute, for example, a hardware specification or software detail. This object is available in API version 60.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Code`	**Type** string **Properties** Create, Filter, Group, idLookup, Nillable, Sort, Update **Description** Code for the attribute definition. This field is unique within your organization.
`CurrencyIsoCode`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Available only if the multicurrency feature is enabled. Contains the ISO code for any currency allowed by the organization. Possible values are: • `BHD`—Bahraini Dinar • `JPY`—Japanese Yen • `USD`—U.S. Dollar The default value is `USD`.
`DataType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort **Description** The data type of the attribute definition. Possible values are: • `Checkbox` • `Currency` • `Date` • `Datetime` • `Multipicklist` • `Number`

Field	Details
	• `Percent` • `Picklist` • `Text`
`DefaultHelpText`	**Type** textarea **Properties** Create, Nillable, Update **Description** The default help text for this attribute.
`DefaultValue`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The default value for this attribute.
`Description`	**Type** textarea **Properties** Create, Nillable, Update **Description** Description of this attribute.
`DeveloperName`	**Type** string **Properties** Create, Filter, Group, Sort **Description** The unique name of the attribute definition record. This name must begin with a letter and use only alphanumeric characters and underscores. It can't include spaces, end with an underscore, or have two consecutive underscores.
`IsActive`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates that the attribute definition is active. Active attributes definitions can be selected for products.

Field	Details
	The default value is `false`.
`IsRequired`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the attribute definition is required for a product. The default value is `false`.
`Label`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** The label for the attribute.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the attribute definition was last referenced.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the attribute definition was last viewed.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** The name of the attribute.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update

Field	Details
	Description The owner of the attribute definition. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User
`PicklistId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort **Description** The ID of the attribute picklist with the valid values for this attribute. This field is a relationship field. **Relationship Name** Picklist **Relationship Type** Lookup **Refers To** AttributePicklist
`SourceSystemIdentifier`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The identifier of the attribute definition in an external system.
`ValueDescription`	**Type** textarea **Properties** Create, Nillable, Update **Description** The default value description for this attribute.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

AttributeDefinitionFeed on page 1673
Feed tracking is available for the object.

AttributeDefinitionHistory on page 1680
History is available for tracked fields of the object.

AttributeDefinitionShare on page 1684
Sharing is available for the object.

BundleBasedAdjustment

Represents the association between the product selling model and the price adjustment for a product or service being sold as a bundle. This object stores information of the product or service's price, the discounts applied, along with its value for a given date range. This object is available in API version 60.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`AdjustmentType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The type of pricing adjustment being made. Possible values are: • `Amount` • `Override` • `Percentage`
`AdjustmentValue`	**Type** double **Properties** Create, Filter, Sort, Update **Description** The type of pricing adjustment being made.

Field	Details
`CurrencyIsoCode`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** ISO code of the currency. Must be one of the valid alphabetic, three-letter currency ISO codes defined by the ISO 4217 standard, such as USD, GBP, or JPY. Must be unique within your organization. Possible values are: • `BHD`—Bahraini Dinar • `JPY`—Japanese Yen • `USD`—U.S. Dollar The default value is `USD`.
`EffectiveFrom`	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** The date and time when the bundle based adjustment comes into effect.
`EffectiveTo`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date and time till when the bundle based adjustment remains effective.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the bundle based adjustment was last referenced.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the bundle based adjustment was last viewed.

Field	Details
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name of the bundle based adjustment record.
`ParentProductId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Represents the immediate parent in the product bundle that the price is configured for. This field is a relationship field. **Relationship Name** ParentProduct **Relationship Type** Lookup **Refers To** Product2
`ParentProductSellingModelId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort **Description** The immediate parent node of a product selling model when nodes are represented in a hierarchial structure. This field is a relationship field. **Relationship Name** ParentProductSellingModel **Relationship Type** Lookup **Refers To** ProductSellingModel
`PriceAdjustmentScheduleId`	**Type** reference **Properties** Create, Filter, Group, Sort

Field	Details
	Description The price adjustment schedule associated with the bundle based adjustment record. This field is a relationship field. **Relationship Name** PriceAdjustmentSchedule **Relationship Type** Lookup **Refers To** PriceAdjustmentSchedule
`PricingTerm`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The number of pricing term units in the pricing term. Used with PricingTermUnit to define the length of the pricing term. For example, if PricingTermUnit is Months and this field is 1, the subscription is priced monthly.If the selling model is one-time, this field must be null.
`PricingTermUnit`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** The unit of time used to define the pricing term. Used with PricingTerm to define the length of the pricing term. For example, if this field is Months and PricingTerm is 1, the subscription is priced monthly. If the selling model is one-time, this field must be null. Possible values are: • `Annual`—Years • `Months` • `Quarterly` • `Semi-Annual`
`ProductId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The ID of the product associated with the product bundle set. This field is a relationship field.

Field	Details
	Relationship Name Product **Relationship Type** Lookup **Refers To** Product2
`ProductSellingModelId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort **Description** The ID of the ProductSellingModel record associated with this bundle based adjustment record. This field is a relationship field. **Relationship Name** ProductSellingModel **Relationship Type** Lookup **Refers To** ProductSellingModel
`RootBundleId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Represents the structural root product in a product's bundle that the price is configured for. This field is a relationship field. **Relationship Name** RootBundle **Relationship Type** Lookup **Refers To** Product2
`RootProductSellingModelId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort

Field	Details
	Description The root product selling model. The primary or base product selling method. This field is a relationship field. **Relationship Name** RootProductSellingModel **Relationship Type** Lookup **Refers To** ProductSellingModel
`SellingModelType`	**Type** picklist **Properties** Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort **Description** Indicates whether the product is sold as a one-time sale, an evergreen subscription, or a subscription with a defined term. Possible values are: • `Evergreen` • `OneTime`—One Time • `TermDefined`—Term-Defined The default value is `OneTime`.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

BundleBasedAdjustmentFeed on page 1673
Feed tracking is available for the object.

BundleBasedAdjustmentHistory on page 1680
History is available for tracked fields of the object.

CostBook

Represents the cost book that contains multiple cost book entries. This object is available in API version 61.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`EffectiveFrom`	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** Date and time when the cost book comes into effect.
`EffectiveTo`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** Date and time till when the cost book is no longer in effect.
`IsDefault`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the cost book is default (`true`) or not (`false`). The default value is `true`.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Timestamp for when the current user last referred to this record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Timestamp for when the current user last viewed a record related to this record.
`Name`	**Type** string

Field	Details
	Properties Create, Filter, Group, idLookup, Sort, Update **Description** Name of the cost book.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** ID of the user who created the record. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

CostBookFeed on page 1673
Feed tracking is available for the object.

CostBookHistory on page 1680
History is available for tracked fields of the object.

CostBookShare on page 1684
Sharing is available for the object.

ContractItemPrice

Represents the price of a product on the contract. This object is available in API version 61.0 and later.

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.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`ContractId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The contract record associated with this contract item price record. This field is a relationship field. **Relationship Name** Contract **Relationship Type** Master-detail **Refers To** Contract (the master object)
`CurrencyIsoCode`	**Type** picklist **Properties** Defaulted on create, Filter, Group, Restricted picklist, Sort **Description** Contains the ISO code for any currency allowed by the organization. Available only if the multicurrency feature is enabled. Valid value is: • `USD`—U.S. Dollar The default value is `USD`.
`DiscountType`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The unit of the discount. Valid values are: • `AdjustmentAmount` • `AdjustmentPercentage`
`DiscountValue`	**Type** double

Field	Details
	Properties Create, Filter, Nillable, Sort, Update **Description** The value of the discount.
`EndDate`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date and time till when the contract item price is no longer in effect.
`ItemId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The ID of the line item record associated with this contract item price record. This field is a polymorphic relationship field. **Relationship Name** Item **Relationship Type** Lookup **Refers To** Product2, ProductCategory
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last referred to this record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed a record related to this record.

Field	Details
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name of the contract item price record.
`Price`	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The unit price of the product that's being sold as part of the contract.
`ProductSellingModelId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ID of the product selling model record associated with this contract item price record. This field is a relationship field. **Relationship Name** ProductSellingModel **Relationship Type** Lookup **Refers To** ProductSellingModel
`SellingModelType`	**Type** picklist **Properties** Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort **Description** Specifies whether the line item is sold as a one-time sale, an evergreen subscription, or a subscription with a defined term. Valid values are: • `Evergreen` • `OneTime` • `TermDefined` The default value is `OneTime`.

Field	Details
`StartDate`	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** The date and time when the contract item price comes into effect.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ContractItemPriceFeed **on page 1673**
Feed tracking is available for the object.

ContractItemPriceHistory **on page 1680**
History is available for tracked fields of the object.

CostBookEntry

Represents the total cost of a product or service that's determined based on various factors that affect a product's price. For example, when a product is manufactured, the weight of the raw material can be a cost factor based on the amount of material required and its shipping cost. This object is available in API version 61.0 and later.

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.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Cost`	**Type** currency **Properties** Create, Filter, Sort, Update **Description** Total cost of the product.
`CostBookId`	**Type** reference

Field	Details
	Properties Create, Filter, Group, Sort **Description** ID of the Cost Book record with which this record is associated. This field is a relationship field. **Relationship Name** CostBook **Relationship Type** Master-detail **Refers To** CostBook (the master object)
`CurrencyIsoCode`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** ISO code for any currency allowed by the organization. Available only for organizations with the multicurrency feature enabled. Valid values are: • `BHD`—Bahraini Dinar • `JPY`—Japanese Yen • `USD`—U.S. Dollar The default value is `USD`.
`Description`	**Type** textarea **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Description of this cost book entry record.
`EffectiveFrom`	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** Date and time when the cost book entry comes into effect.
`EffectiveTo`	**Type** dateTime

Field	Details
	Properties Create, Filter, Nillable, Sort, Update **Description** Date and time till when the cost book entry is no longer in effect.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Timestamp for when the current user last referred to this record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Timestamp for when the current user last viewed a record related to this record.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Name of the cost book entry.
`ProductId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Required. ID of the Product2 record with which this record is associated. This field must be specified when creating Product2 records. This field can't be changed in an update. This field is a relationship field. **Relationship Name** Product **Relationship Type** Lookup **Refers To** Product2

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

CostBookEntryFeed on page 1673
Feed tracking is available for the object.

CostBookEntryHistory on page 1680
History is available for tracked fields of the object.

PriceAdjustmentSchedule

Represents a series of tiered discounts based on the number of items purchased. This object is available in API version 60.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`AdjustmentMethod`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update **Description** The method for applying tiered pricing. Possible values are: • Range—All items receive the discount of the highest tier the quantity falls in. • Slab—Items receive the discount defined for the tier they fall in. Possible values are: • `Range` • `Slab` The default value is `Range`.
`CurrencyIsoCode`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Available only if the multicurrency feature is enabled. Contains the ISO code for any currency allowed by the organization. Possible values are:

Field	Details
	• `BHD`—Bahraini Dinar • `JPY`—Japanese Yen • `USD`—U.S. Dollar The default value is `USD`.
`Description`	**Type** textarea **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Text description of the price adjustment schedule.
`EffectiveFrom`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date and time when the price adjustment schedule comes into effect.
`EffectiveTo`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date and time when the price adjustment schedule remains effective.
`IsActive`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the price adjustment schedule is active (true) or not (false). You can change this field's value as often as necessary. Label is **Active**. Default value is *`False`*.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Indicates whether the price adjustment schedule has been archived (true) or not (false). This field is read-only.

Field	Details
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed this record. If this value is null, it's possible that this record was referenced (LastReferencedDate) and not viewed.
Name	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** Required. The name of the price adjustment schedule. This field is read-only. Label is Price Adjustment Schedule Name.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The Salesforce ID of the sales representative who owns the price adjustment schedule. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User
Pricebook2Id	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The price book associated with this price adjustment schedule record. This field is a relationship field. **Relationship Name** Pricebook2 **Relationship Type** Lookup

Field	Details
	Refers To Pricebook2
`ScheduleType`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update **Description** The type of price adjustment schedule. Possible values are: • `Attribute` • `Bundle` • `Custom` • `Term` • `Volume` The default value is `Volume`.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

PriceAdjustmentScheduleShare on page 1684
Sharing is available for the object.

PriceAdjustmentTier

Represents a discount tier in a price adjustment schedule. This object is available in API version 60.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`AdjustmentType`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field	Details
	Description The type of price adjustment.
`CurrencyIsoCode`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Available only if the multicurrency feature is enabled. Contains the ISO code for any currency allowed by the organization. Possible values are: • `BHD`—Bahraini Dinar • `JPY`—Japanese Yen • `USD`—U.S. Dollar The default value is `USD`.
`EffectiveFrom`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date and time when the price adjustment tier comes into effect.
`EffectiveTo`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date and time when the price adjustment tier remains effective.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Indicates when the user last referred to this record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort

Field	Details
	Description The timestamp for when the current user last viewed this record. If this value is null, it's possible that this record was referenced (LastReferencedDate) and not viewed.
`LowerBound`	**Type** double **Properties** Create, Filter, Sort, Update **Description** The minimum quantity the discount can be applied to. It must be a positive integer and less than or equal to the upper bound of the tier.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name of the price adjustment tier.
`PriceAdjustmentScheduleId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The ID of the price adjustment schedule that the discount is applied to. This field is a relationship field. **Relationship Name** PriceAdjustmentSchedule **Relationship Type** Lookup **Refers To** PriceAdjustmentSchedule
`PricingTerm`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The number of pricing term units in the pricing term. Used with PricingTermUnit to define the length of the pricing term. For example, if PricingTermUnit is Months and this field is 1, the subscription is priced monthly.If the selling model is one-time, this field must be null.

Field	Details
`PricingTermUnit`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** The unit of time used to define the pricing term. Used with PricingTerm to define the length of the pricing term. For example, if this field is Months and PricingTerm is 1, the subscription is priced monthly. If the selling model is one-time, this field must be null. Possible values are: • `Annual`—Years • `Months`
`Product2Id`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ID of the product associated with the price adjustment tier. This field is a relationship field. **Relationship Name** Product2 **Relationship Type** Lookup **Refers To** Product2
`ProductSellingModelId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ID of the ProductSellingModel record associated with this price adjustment tier record. This field is a relationship field. **Relationship Name** ProductSellingModel **Relationship Type** Lookup **Refers To** ProductSellingModel

Field	Details
`SellingModelType`	**Type** picklist **Properties** Defaulted on create, Filter, Group, Restricted picklist, Sort **Description** Indicates whether the product is sold as a one-time sale, an evergreen subscription, or a subscription with a defined term. Possible values are: • `Evergreen` • `OneTime`—One Time • `TermDefined`—Term-Defined The default value is `OneTime`.
`TierType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The unit of the discount. Possible values are: • AdjustmentAmount—An amount discounted from an item's list price • AdjustmentPercentage—A percentage discounted from an item's list price Possible values are: • `AdjustmentAmount`—Amount • `AdjustmentPercentage`—Percentage • `OverrideAmount`—Override
`TierValue`	**Type** double **Properties** Create, Filter, Sort, Update **Description** The value of the discount.
`UpperBound`	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The maximum quantity the discount can be applied to. Must be a positive integer. Not inclusive. Set this value one digit higher than the quantity you want the tier to include. For

Field	Details
	example, if a tier's upper bound is 99, set the value of UpperBound to 100. For the last tier, the value is optional.

Usage

To use PriceAdjustmentTiers, associate them with a PriceAdjustmentSchedule.

Tiers can't overlap, and no gaps are allowed between tiers.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

PriceAdjustmentTierFeed on page 1673
Feed tracking is available for the object.

PriceAdjustmentTierHistory on page 1680
History is available for tracked fields of the object.

PriceBook2

Represents a price book that contains the list of products that your org sells. This object is available in API version 60.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`CurrencyIsoCode`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Available only if the multicurrency feature is enabled. Contains the ISO code for any currency allowed by the organization. Possible values are: • `BHD`—Bahraini Dinar • `JPY`—Japanese Yen • `USD`—U.S. Dollar The default value is `USD`.

Field	Details
`Description`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Text description of the price book.
`IsActive`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the price book is active (true) or not (false). Inactive price books are hidden in many areas in the user interface. You can change this field's value as often as necessary. Label is **Active**. The default value is `false`.
`IsArchived`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Indicates whether the price book has been archived (true) or not (false). This field is read only. The default value is `false`.
`IsStandard`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Indicates whether the price book is the standard price book for the org (true) or not (false). Every org has one standard price book—all other price books are custom price books. The default value is `false`.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed a record related to this record.

Field	Details
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed this record. If this value is null, it's possible that this record was referenced (LastReferencedDate) and not viewed.
Name	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** Required. Name of this object. This field is read-only for the standard price book.
ValidFrom	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date and time when a price book is initially valid. If this field is null, the price book is valid immediately when active.
ValidTo	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date and time when a price book is valid to. If this field is null, the price book is valid until it's deactivated.

Usage

A price book is a list of products that your org sells.

- Each org has one standard price book that defines the standard or generic list price for each product or service that it sells.
- An org can have multiple custom price books to use for specialized purposes, such as for discounts, different channels or markets, or select accounts or opportunities. While your client application can create, delete, and update custom price books, your client application can only update the standard price book.
- For some orgs, the standard price book is the only price needed. If you set up other price books, you can reference the standard price book when setting up list prices in custom price books.

Use this object to query standard and custom price books that have been configured for your org. A common use of this object is to allow your client application to obtain valid Pricebook2 object IDs for use when configuring PricebookEntry records via the API.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

PriceBook2ChangeEvent on page 1671
Change events are available for the object.

PriceBook2History on page 1680
History is available for tracked fields of the object.

PriceBookEntry

Represents a product entry (an association between a Pricebook2 and Product2) in a price book. This object is available in API version 60.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Note: Salesforce Object Search Language (SOSL) allows you to search records across standard and custom objects. When filtering records in the PriceBookEntry object using SOSL, you can only sort by fields related to Product2.

Field	Details
`ActivePriceAdjustmentQuantity`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The count of active price adjustment schedules associated with the price book entry.
`CurrencyIsoCode`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort **Description** Available only for organizations with the multicurrency feature enabled. Contains the ISO code for any currency allowed by the organization. Possible values are: • `BHD`—Bahraini Dinar

Field	Details
	• `JPY`—Japanese Yen • `USD`—U.S. Dollar The default value is `USD`.
`IsActive`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether this price book entry is active (true) or not (false). Although you can never delete PricebookEntry records, your client application can set this flag to false. Inactive PricebookEntry records are hidden in many areas in the user interface. You can change this flag on a PricebookEntry record as often as necessary. The default value is `false`.
`IsArchived`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Indicates whether the PricebookEntry has been archived (true) or not (false). This field is set to true when the Product2 record it's associated with is archived, or when the Pricebook2 record is archived. This field is read only. The default value is `false`.
`IsDerived`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort **Description** Indicates whether the product's price was derived (yes) or not (no). The default value is `false`.
`Name`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** Name of this price book entry record. This read-only field references the value in the Name field of the Product2 record.

Field	Details
`Pricebook2Id`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Required. ID of the Pricebook2 record with which this record is associated. This field must be specified when creating Pricebook2 records. It can't be changed in an update. This field is a relationship field. **Relationship Name** Pricebook2 **Relationship Type** Lookup **Refers To** Pricebook2
`Product2Id`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Required. ID of the Product2 record with which this record is associated. This field must be specified when creating Product2 records. It can't be changed in an update. This field is a relationship field. **Relationship Name** Product2 **Relationship Type** Lookup **Refers To** Product2
`ProductCode`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** Product code for this record. This read-only field references the value in the ProductCode field of the associated Product2 record.
`ProductSellingModelId`	**Type** reference

Field	Details
	Properties Create, Filter, Group, Nillable, Sort **Description** The ID of the related product selling model. This field is a relationship field. **Relationship Name** ProductSellingModel **Relationship Type** Lookup **Refers To** ProductSellingModel
`UnitPrice`	**Type** currency **Properties** Create, Filter, Sort, Update **Description** Required. Unit price for this price book entry. You can specify a value only if UseStandardPrice is set to false.
`UseStandardPrice`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether this price book entry uses the standard price defined in the standard Pricebook2 record (true) or not (false). If set to true, then the UnitPrice field is read-only, and the value is the same as the UnitPrice value in the corresponding PricebookEntry in the standard price book (that is, the PricebookEntry record whose Pricebook2Id refers to the standard price book and whose Product2Id and CurrencyIsoCode are the same as this record). For PricebookEntry records associated with the standard Pricebook2 record, this field must be set to true. The default value is `false`.

Usage

Use this object to define the association between your organization's products (Product2) and your organization's standard price book or to custom price books (Pricebook2). Create one PricebookEntry record for each standard or custom price and currency combination for a product in a Pricebook2.

When creating these records, you must specify the IDs of the associated Pricebook2 record and Product2 record. Once these records are created, your client application can't update these IDs.

This object is defined only for those organizations that have products enabled as a feature. If the organization doesn't have the products feature enabled, then the PricebookEntry object doesn't appear in the describeGlobal call, and you can't access it.

If you delete a PriceBookEntry that is referenced by a line item, the line item is unaffected, but the PriceBookEntry is archived and unavailable from the API. Deleted PriceBookEntry records can't be recovered.

You must load the standard price for a product before you're permitted to load its custom prices.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

PriceBookEntryChangeEvent **on page 1671**
Change events are available for the object.

PriceBookEntryHistory **on page 1680**
History is available for tracked fields of the object.

PriceBookEntryDerivedPrice

Represents the price of a product that's derived from another source such as a product or an asset. This object is available in API version 61.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`ContributingProductId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort **Description** Source product from which the derived price is calculated. The source product is associated with the derived price product. This field is a relationship field. **Relationship Name** ContributingProduct **Relationship Type** Lookup **Refers To** Product2

Field	Details
`CurrencyIsoCode`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** ISO code for any currency allowed by the organization. Available only if the multicurrency feature is enabled. Valid values are: • `BHD`—Bahraini Dinar • `JPY`—Japanese Yen • `USD`—U.S. Dollar The default value is `USD`.
`DerivedPricingScope`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Scope of the product based on which the derived price is calculated. Valid values are: • `Both` • `NonTransactional` • `Transactional`
`EffectiveFrom`	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** Date and time when the derived pricing comes into effect.
`EffectiveTo`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** Date and time when the derived pricing is no longer in effect.
`Formula`	**Type** string

Field	Details
	Properties Create, Filter, Sort, Update **Description** Coded format of the formula used to calculate the derived price of a product from another product or asset.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Timestamp for when the current user last referred to this record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Timestamp for when the current user last viewed this record. If this value is null, it's possible that this record was referenced (`LastReferencedDate`) and not viewed.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Name of the derived price record.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** ID of the user who created the record. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User

Field	Details
`PricebookEntryId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Price book entry associated with the source product. This field is a relationship field. **Relationship Name** PricebookEntry **Relationship Type** Lookup **Refers To** PricebookEntry
`PricebookId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort **Description** Price book associated with the source product. This field is a relationship field. **Relationship Name** Pricebook **Relationship Type** Lookup **Refers To** Pricebook2
`PricingSource`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort **Description** Pricing type used to calculate the derived price of the product. Valid values are: • `Header` • `Product`
`ProductId`	**Type** reference

Field	Details
	Properties Filter, Group, Sort **Description** Product associated with the derived pricing. This field is a relationship field. **Relationship Name** Product **Relationship Type** Lookup **Refers To** Product2
`ProductSellingModelId`	**Type** reference **Properties** Filter, Group, Sort **Description** Product selling model associated with this record. This field is a relationship field. **Relationship Name** ProductSellingModel **Relationship Type** Lookup **Refers To** ProductSellingModel

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

PriceBookEntryDerivedPriceFeed on page 1673
Feed tracking is available for the object.

PriceBookEntryDerivedPriceHistory on page 1680
History is available for tracked fields of the object.

PriceBookEntryDerivedPriceShare on page 1684
Sharing is available for the object.

PricingAdjBatchJob

Represents the collective update of multiple records on their prices and other adjustments. This object is available in API version 62.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`EffectiveFrom`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date and time when the updated value can be considered for a pricing adjustment batch job.
`EffectiveTo`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date and time until when the updated value is effective and can be considered for a pricing adjustment batch job.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Indicates whether the pricing adjustment batch job has been archived (`true`) or not (`false`). This field is read-only.
`LastTriggeredDate`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date and time when the pricing adjustment batch job was last triggered.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort

Field	Details
	Description The timestamp for when the current user last viewed this record. If this value is null, it's possible that this record was referenced (`LastReferencedDate`) and not viewed.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name of the pricing adjustment batch job.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The Salesforce ID of the sales representative who owns the pricing procedure resolution. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
`ProcessedRecordsCount`	**Type** long **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The total number of records that were successfully updated.
`RecordCount`	**Type** long **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The total number of records that have been processed.
`RecordList`	**Type** textarea **Properties** Create

Field	Details
	Description The list of record IDs eligible for a pricing adjustment batch job.
`ShouldSkipBulkRetry`	**Type** boolean **Properties** Create, Filter, Group, Nillable, Sort **Description** For internal use only.
`Status`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The processing status of the pricing adjustment batch job. Valid values are: • `Completed` • `Failed` • `InProgress` • `New` • `PartiallyCompleted` • `Rerun`
`TargetObject`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort **Description** The target object of the pricing adjustment batch job. Valid values are: • `AttributeBasedAdjustment` • `BundleBasedAdjustment` • `PriceAdjustmentTier` • `PricebookEntry`
`UpdateType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update

Field	Details
	Description The type of update being made by the pricing adjustment batch job. Valid values are: • `Amount` • `Override` • `Percentage`
`UpdateValue`	**Type** double **Properties** Create, Filter, Sort, Update **Description** The numerical value of the update.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

PricingAdjBatchJobFeed on page 1673
Feed tracking is available for the object.

PricingAdjBatchJobHistory on page 1680
History is available for tracked fields of the object.

PricingAdjBatchJobShare on page 1684
Sharing is available for the object.

PricingAdjBatchJobLog

Represents the report that contains a list of failed adjustment requests along with an error message that describes the reason for failure. This object is available in API version 62.0 and later.

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.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
AdjustedValue	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The adjusted value of a record. The stored value is used even if another pricing adjustment batch job is triggered again.
EffectiveFrom	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date and time when the failed versioned record is generated. This is only applicable for Price Adjustment records.
EffectiveTo	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date and time until when the failed versioned record is available. This is applicable only for Price Adjustment records.
ErrorCode	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The error code for the failure during the record update process.
ErrorMessage	**Type** textarea **Properties** Create, Update **Description** The error message that's generated for the failure during the record update process.
LastReferencedDate	**Type** dateTime

Field	Details
	Properties Filter, Nillable, Sort **Description** Indicates whether the pricing adjustment batch job has been archived (`true`) or not (`false`). This field is read-only.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed this record. If this value is null, it's possible that this record was referenced (`LastReferencedDate`) and not viewed.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. The name of the pricing adjustment batch job.
`PricingAdjBatchJobId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The pricing adjustment batch job associated with the pricing adjustment batch job log. This field is a relationship field. **Relationship Name** PricingAdjBatchJob **Relationship Type** Master-detail **Refers To** PricingAdjBatchJob (the master object)
`TargetRecord`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** The ID of the record for which a pricing adjustment error was generated.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

PricingAdjBatchJobLogFeed
Feed tracking is available for the object.

PricingAdjBatchJobLogHistory **on page 1680**
History is available for tracked fields of the object.

PricingAPIExecution

Represents the pricing resolution for an pricing element determined using strategy name and formula. This object is available in API version 63.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`ApiEndpoint`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The unique API endpoint that is called.
`ApiType`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update **Description** Specifies the API type of the pricing API execution. Possible values are: • `NGP` The default value is `NGP`.
`CurrencyIsoCode`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field	Details
	Description Available only if the multicurrency feature is enabled. Contains the ISO code for any currency allowed by the organization. Possible values are: • `BHD`—Bahraini Dinar • `JPY`—Japanese Yen • `USD`—U.S. Dollar The default value is `USD`.
`ExecutionKey`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** The unique execution ID generated each time a pricing API runs.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed a record related to this record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed this record. If this value is null, it's possible that this record was referenced (LastReferencedDate) and not viewed.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name of the record.
`OwnerId`	**Type** reference

Field	Details
	Properties Create, Defaulted on create, Filter, Group, Sort, Update **Description** The Salesforce ID of the sales representative who owns the pricing procedure resolution. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
`ReferenceKey`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Optional. The reference ID that a consuming workstream must pass in the API to search for specific logs in the Pricing Operations Console.
`Status`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update **Description** The status of the API response. Possible values are: • `Failure` • `Partial_Success`—Partial Success • `Success` The default value is `Success`.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

PricingAPIExecutionFeed on page 1673
Feed tracking is available for the object.

PricingAPIExecutionHistory on page 1680
History is available for tracked fields of the object.

PricingAPIExecutionShare on page 1684
Sharing is available for the object.

PricingProcedureResolution

Represents a selection for a pricing procedure to execute a pricing process from a list of pricing procedures available. This object is available in API version 60.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`CurrencyIsoCode`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Available only if the multicurrency feature is enabled. Contains the ISO code for any currency allowed by the organization. Possible values are: • `BHD`—Bahraini Dinar • `JPY`—Japanese Yen • `USD`—U.S. Dollar The default value is `USD`.
`EffectiveFrom`	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** The date and time when the pricing procedure resolution comes into effect.
`EffectiveTo`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date and time till when the pricing procedure resolution remains effective.
`LastReferencedDate`	**Type** dateTime

Field	Details
	Properties Filter, Nillable, Sort **Description** Indicates whether the pricing procedure resolution has been archived (true) or not (false). This field is read-only.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed this record. If this value is null, it's possible that this record was referenced (LastReferencedDate) and not viewed.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. The name of the pricing procedure resolution.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The Salesforce ID of the sales representative who owns the pricing procedure resolution. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User
`PricingProcedureId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The pricing procedure record associated with this pricing procedure resolution.

Field	Details
	This field is a relationship field. **Relationship Name** PricingProcedure **Relationship Type** Lookup **Refers To** ExpressionSet
`ProcedureType`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The pricing data store associated with this pricing recipe field mappings.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

PricingProcedureResolutionFeed on page 1673
Feed tracking is available for the object.

PricingProcedureResolutionHistory on page 1680
History is available for tracked fields of the object.

PricingProcedureResolutionShare on page 1684
Sharing is available for the object.

PricingProcessExecution

Represents a record generated during the execution of a discovery or pricing procedure. Multiple procedures may be performed within a single API call, with each recorded in a Pricing API Execution record. This object is available in API version 63.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`CurrencyIsoCode`	**Type** picklist

Field	Details
	Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Available only if the multicurrency feature is enabled. Contains the ISO code for any currency allowed by the organization. Possible values are: • `BHD`—Bahraini Dinar • `JPY`—Japanese Yen • `USD`—U.S. Dollar The default value is `USD`.
`ExecutionKey`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** The unique execution ID generated each time a pricing API runs.
`ExecutionType`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update **Description** The type of execution defined internally within the pricing API. Possible values are: • `Api_Execution`—Api Execution • `Discovery` • `Discovery_Line`—Discovery Line • `Pricing` • `Pricing_Line`—Pricing Line The default value is `Pricing`.
`ExecutionTypeKey`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** The unique execution type ID generated internally for procedure executions, such as pricing or discovery procedures.

Field	Details
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed a record related to this record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed this record. If this value is null, it's possible that this record was referenced (LastReferencedDate) and not viewed.
Message	**Type** textarea **Properties** Create, Nillable, Update **Description** The message generated upon running a pricing process.
Name	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name of the record.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The Salesforce ID of the sales representative who owns the pricing procedure resolution. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User

Field	Details
`Status`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update **Description** Specifies the status of the execution type. Possible values are: • `Failure` • `Partial_Success`—Partial Success • `Success` The default value is `Success`.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

PricingProcessExecutionFeed on page 1673
Feed tracking is available for the object.

PricingProcessExecutionHistory on page 1680
History is available for tracked fields of the object.

PricingProcessExecutionShare on page 1684
Sharing is available for the object.

ProductPriceHistoryLog

Stores historical pricing data based on the product's price range. This object is available in API version 62.0 and later.

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.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`CurrencyIsoCode`	**Type** picklist

Field	Details
	Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** ISO code for any currency allowed by the organization. Available only for organizations with the multicurrency feature enabled. Valid values are: • `BHD`—Bahraini Dinar • `JPY`—Japanese Yen • `USD`—U.S. Dollar The default value is `USD`.
`Date`	**Type** date **Properties** Create, Filter, Group, Sort, Update **Description** The date when the product price history log record is created.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Indicates whether the product price history log has been archived (`true`) or not (`false`). This field is read-only.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed this record. If this value is null, it's possible that this record was referenced (`LastReferencedDate`) and not viewed.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. The name of the product price history log.

Field	Details
`ProductPriceRangeId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The product price range associated with this product price history log record. This field is a relationship field. **Relationship Name** ProductPriceRange **Relationship Type** Master-detail **Refers To** ProductPriceRange (the master object)
`TrackedPrice`	**Type** currency **Properties** Create, Filter, Sort, Update **Description** The price for a product recorded for a particular date.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ProductPriceHistoryLogFeed on page 1673
Feed tracking is available for the object.

ProductPriceHistoryLogHistory on page 1680
History is available for tracked fields of the object.

ProductPriceRange

Represents the price range of a product determined by using a product selling model that's stored in the relevant price book. This object is available in API version 62.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`CurrencyIsoCode`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** ISO code for any currency allowed by the organization. Available only for organizations with the multicurrency feature enabled. Valid values are: • `BHD`—Bahraini Dinar • `JPY`—Japanese Yen • `USD`—U.S. Dollar The default value is `USD`.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Indicates whether the product price range has been archived (`true`) or not (`false`). This field is read-only.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed this record. If this value is null, it's possible that this record was referenced (`LastReferencedDate`) and not viewed.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. The name of the product price range.
`OwnerId`	**Type** reference

Field	Details
	Properties Create, Defaulted on create, Filter, Group, Sort, Update **Description** The Salesforce ID of the sales representative who owns the product price range. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
`ProductId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The product for which the price range is being determined. This field is a relationship field. **Relationship Name** Product **Refers To** Product2

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ProductPriceRangeFeed on page 1673
Feed tracking is available for the object.

ProductPriceRangeHistory on page 1680
History is available for tracked fields of the object.

ProductPriceRangeShare on page 1684
Sharing is available for the object.

ProductSellingModel

Defines one method by which a product can be sold; for example, as a one-time sale, an evergreen subscription, or a termed subscription. If the product is sold on subscription, this object defines the subscription's term. A product can have multiple product selling models. This object is available in API version 60.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Indicates when the user last referred to this record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Indicates when the user last viewed this record.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** The name given to the product selling model.
`PricingTerm`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The number of pricing term units in the pricing term. Used with PricingTermUnit to define the length of the pricing term. For example, if PricingTermUnit is Months and this field is 1, the subscription is priced monthly.If the selling model is one-time, this field must be null.
`PricingTermUnit`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field	Details
	Description The unit of time used to define the pricing term. Used with PricingTerm to define the length of the pricing term. For example, if this field is Months and PricingTerm is 1, the subscription is priced monthly. If the selling model is one-time, this field must be null. Possible values are: • `Annual`—Years • `Months`
`SellingModelType`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update **Description** Indicates whether the product is sold as a one-time sale, an evergreen subscription, or a subscription with a defined term. Possible values are: • `Evergreen` • `OneTime`—One Time • `TermDefined`—Term-Defined The default value is `OneTime`.
`Status`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update **Description** The status of the product selling model. Possible values are: • `Active` • `Draft` • `Inactive` The default value is `Draft`.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ProductSellingModelFeed on page 1673
Feed tracking is available for the object.

ProductSellingModelHistory on page 1680
History is available for tracked fields of the object.

ProductSellingModelDataTranslation

Represents the translated values of the data stored within the ProductSellingModel record's fields. This object is available in API version 61.0 and later.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

- Your organization must be using Enterprise, Unlimited, or Developer edition.
- Translation Workbench and data translation must be enabled in your org.
- To view this object, you must have the "View Setup and Configuration" permission.

Fields

Field	Details
`IsOutOfDate`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Indicates whether the translation is out-of-date (`true`) or current (`false`). A translation is out-of-date if the parent Product2 record is updated after the last translation was filed.
`Language`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort **Description** The language for these translated values.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update

Field	Details
	Description The translated value for the ProductSellingModel record name. This field is required to translate the text in other fields.
`ParentId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The ID of the ProductSellingModel record associated with the data that's being translated. This field is a relationship field. **Relationship Name** Parent **Relationship Type** Lookup **Refers To** ProductSellingModel

Usage

Use this object to translate the data stored in a ProductSellingModel record into the different languages supported by Salesforce. If data translation is enabled for custom fields on the ProductSellingModel object, additional ProductSellingModelDataTranslation fields exist for translating the data contained within those fields.

You can't use a custom external id field in an upsert call for a ProductSellingModelDataTranslation object.

ProductSellingModelOption

A junction object between Product Selling Model and Product2. This object is available in API version 60.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Description`	**Type** textarea **Properties** Create, Nillable, Update

Field	Details
	Description The description of the product selling model option.
`DisplayName`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name of the product selling model option to display to customers.
`Increment`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The number of pricing term units that can be used to increase a subscription term.
`IsDefault`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indcates the default product selling model for a product. Setting a default is optional. A product can only have one default product selling model. The default value is `false`. This field requires Industries EPC.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed this record, a record related to this record, or a list view.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed this record or list view. If this value is null, the user might have only accessed this record or list view but not viewed it.

Field	Details
`Maximum`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The maximum number of pricing term units for a subscription term.
`Minimum`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The minimum number of pricing term units for a subscription term.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name of the product selling model option.
`Product2Id`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The ID of the Product2 record associated with this ProductSellingModelOption record. This field is a relationship field. **Relationship Name** Product2 **Relationship Type** Lookup **Refers To** Product2
`ProductSellingModelId`	**Type** reference **Properties** Create, Filter, Group, Sort

Field	Details
	Description The ID of the ProductSellingModel record associated with this ProductSellingModelOption record. This field is a relationship field. **Relationship Name** ProductSellingModel **Relationship Type** Lookup **Refers To** ProductSellingModel
`ProrationPolicyId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort **Description** The ID of the ProrationPolicy record associated with this ProductSellingModelOption record. This field is a relationship field. **Relationship Name** ProrationPolicy **Relationship Type** Lookup **Refers To** ProrationPolicy

Salesforce Pricing Business APIs

Perform pricing request, create context instance, sync pricing data, and manage pricing recipes and pricing waterfall details by using the Salesforce Pricing Business APIs.

This table lists the available Salesforce Pricing resources.

Resource	Description
`/connect/core-pricing/price-contexts/`***`contextid`*** (POST)	Perform a pricing request by using the instance ID of a context.
`/connect/core-pricing/pricing` (POST)	Create and hydrate context instance in a single request. Provide a comprehensive response that contains final

Resource	Description
	pricing details per line items and related errors, if any.
`/connect/core-pricing/sync/`***`pricingSyncOrigin`*** (GET)	Sync pricing data to ensure that the lookup tables contain the latest pricing data.
`/connect/core-pricing/recipe` (GET)	Get the mapping details of pricing recipes to the associated pricing recipe table.
`/connect/core-pricing/recipe/mapping` (POST)	Create a mapping between the pricing recipe and the Decision Tables. Post recipes with lookup tables or procedures.
`/connect/core-pricing/versioned-revise-details` (POST)	Create revisions of a pricing request with versions for adjustment entities.
`/connect/core-pricing/waterfall/`***`lineItemId`***`/`***`executionId`*** (GET)	Get the persisted price waterfall that stores the process logs. Price waterfall provides insights into every step of the pricing process.
`/connect/core-pricing/waterfall` (POST)	Create a log of price waterfall. Price waterfall provides insights into every step of the pricing process.
`/connect/core-pricing/pbeDerivedPricingSourceProduct` (POST)	Get the source product for the Price Book Entry (PBE) derived pricing.
/connect/core-pricing/apiexecutionlogs/*`executionId`* (GET)	Get the log details of a pricing API execution record by using the execution ID.
/connect/core-pricing/pricing-process-execution/*`executionId`* (GET)	Get the execution details of a pricing process by using the execution ID.
/connect/core-pricing/pricing-process-execution/lineitems/*`executionId`*/*`executionType`* (GET)	Get the pricing execution details for the line items of a pricing process by using the execution ID and execution type.

This section lists the available Procedure Plan Definition-related resources. Use procedure plan definitions to define criteria for all pricing process-related requirements in one central location, and to set up the procedures based on these requirements.

Resource	Description
`/connect/procedure-plan-definitions` (GET, POST)	Get the records of procedure plan definitions. Additionally, create a record of a procedure plan definition.
`/connect/procedure-plan-definitions/`***`procedurePlanDefinitionId`***(GET, PATCH, DELETE)	Get, update, or delete a procedure plan definition record by using the record ID.
`/connect/procedure-plan-definitions/evaluate` (POST)	Evaluate a procedure plan definition based on a primary object to check for prerequisites such as usage type and context mapping details.
`/connect/procedure-plan-definitions/evaluate/`***`procedurePlanDefinitionName`*** (POST)	Evaluate a procedure plan definition based on the name of a definition to check for prerequisites such as usage type and context mapping details.
`/connect/procedure-plan-definitions/`***`procedurePlanDefinitionId`***`/version` (POST)	Create records of a procedure plan version with details.
`/connect/procedure-plan-definitions/versions/`***`procedurePlanVersionId`*** (GET, PATCH, DELETE)	Get, update, or delete a procedure plan definition version record by using the record ID.

Resources
Learn more about the available Salesforce Pricing resources.

Request Bodies
Learn more about the available Salesforce Pricing API request bodies.

Response Bodies
Learn more about the available Salesforce Pricing API response bodies.

SEE ALSO:

Connect REST API Developer Guide: Introduction

Resources

Learn more about the available Salesforce Pricing resources.

PBE Derived Pricing (POST)
Get the source product for the Price Book Entry (PBE) derived pricing.

Price Context (POST)

Perform a pricing request by using the instance ID of a context.

Pricing (POST)

Create and hydrate context instance in a single request. Provide a comprehensive response that contains final pricing details per line items and related errors, if any.

API Execution Logs (GET)

Get the log details of a pricing API execution record by using the execution ID.

Pricing Process Execution (GET)

Get the execution details of a pricing process by using the execution ID.

Pricing Process Execution for Line Items (GET)

Get the pricing execution details for the line items of a pricing process by using the execution ID and execution type.

Pricing Data Sync (GET)

Sync pricing data to ensure that the lookup tables contain the latest pricing data.

Pricing Recipe (GET)

Get the mapping details of pricing recipes to the associated pricing recipe table.

Pricing Recipe Mapping (POST)

Create a mapping between the pricing recipe and the Decision Tables. Post recipes with lookup tables or procedures.

Pricing Versioned Revision Details (POST)

Create revisions of a pricing request with versions for adjustment entities.

Pricing Waterfall (GET)

Get the persisted price waterfall that stores the process logs. Price waterfall provides insights into every step of the pricing process.

Pricing Waterfall (POST)

Create a log of price waterfall. Price waterfall provides insights into every step of the pricing process.

Procedure Plan Definitions (GET, POST)

Get the records of procedure plan definitions. Additionally, create a record of a procedure plan definition.

Procedure Plan Definition By ID (GET, PATCH, DELETE)

Get, update, or delete a procedure plan definition record by using the record ID.

Procedure Plan Evaluation By Object (POST)

Evaluate a procedure plan definition based on a primary object to check for prerequisites such as usage type and context mapping details.

Procedure Plan Evaluation By Definition Name (POST)

Evaluate a procedure plan definition based on the name of a definition to check for prerequisites such as usage type and context mapping details.

Procedure Plan Version (POST)

Create records of a procedure plan version with details.

Procedure Plan Version Details (GET, PATCH, DELETE)

Get, update, or delete a procedure plan definition version record by using the record ID.

PBE Derived Pricing (POST)

Get the source product for the Price Book Entry (PBE) derived pricing.

Resource

```
/connect/core-pricing/pbeDerivedPricingSourceProduct
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/core-pricing/pbeDerivedPricingSourceProduct
```

Available version

61.0

HTTP methods

POST

Request body for POST

JSON example

```
{
"productId":"01txx0000006i2SAAQ",
"pricebookEntryId":"01uxx0000008yYcAAI",
"effectiveFrom":"2020-01-01T22:53:20.000Z",
"effectiveTo":"2021-01-01T22:53:20.000Z"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
effectiveFrom	String	Date from when the price book entry is effective.	Required	61.0
effectiveTo	String	Date until when the price book entry is effective.	Required	61.0
pricebookEntryId	String	ID of the price book entry.	Required	61.0
productId	String	ID of the price book.	Required	61.0

Response body for POST

PBE Derived Pricing

Price Context (POST)

Perform a pricing request by using the instance ID of a context.

Resource

```
/connect/core-pricing/price-contexts/contextid
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/core-pricing/price-contexts/0U3RM00000000SR0AY
```

Available version

60.0

HTTP methods

POST

Request body for POST

JSON example

```
{
    "configurationOverrides": {
       "skipWaterfall": true,
       "useSessionScopedContext": true,
       "persistContext": true,
       "taggedData": false
    }
    "procedureName": "ES1"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`configurationOverrides`	Configuration Override Input	Parameters to override pricing configuration.	Optional	60.0
`procedureName`	String	Name of the pricing procedure.	Optional	60.0

Response body for POST

Pricing Response

Pricing (POST)

Create and hydrate context instance in a single request. Provide a comprehensive response that contains final pricing details per line items and related errors, if any.

Resource

```
/connect/core-pricing/pricing
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/core-pricing/pricing
```

Available version

60.0

Requires Chatter

No

HTTP methods

POST

Request body for POST

JSON example

```
{
  "contextDefinitionId": "11Oxx0000006PdxEAE",
  "contextMappingId": "11jxx0000004LDDAA2",
  "jsonDataString": {
    "Cart": [
      {
        "id": "cart_1001",
        "cart_id": "cart_1001",
        "PriceBookId": "PriceBookId_1001",
        "businessObjectType": "Cart",
        "CartItem": [
          {
            "id": "lineItem_1001",
            "line_item_id": "lineItem_1001",
            "Quantity": 7,
            "PriceType": "OneTime",
            "Frequency": "",
            "UOM": "",
            "businessObjectType": "CartItem",
            "product_id": "01txx0000006i44AAA",
            "UnitPrice": 6.8,
            "NetUnitPrice": 0,
            "Attribute": [
              {
                "name": "Color",
                "code": "RED",
                "isPriceImpacting": true,
                "businessObjectType": "Attribute",
                "id": "Attribute_1001",
                "attribute_id": "Attribute_1001"
              },
              {
                "name": "Size",
                "code": "10INCH",
                "isPriceImpacting": true,
                "businessObjectType": "Attribute",
                "id": "Attribute_1002",
                "attribute_id": "Attribute_1002"
              }
            ]
          },
          {
            "id": "lineItem_1002",
            "line_item_id": "lineItem_1002",
            "quantity": 3,
            "PriceType": "OneTime",
            "Frequency": "",
            "UOM": "",
            "businessObjectType": "CartItem",
            "product_id": "01txx0000006i2SAAQ",
            "unitprice": 6,
```

```
            "NetUnitPrice": 0,
            "Attribute": [
              {
                "name": "Color",
                "code": "BLUE",
                "isPriceImpacting": true,
                "businessObjectType": "Attribute",
                "id": "Attribute_1003",
                "attribute_id": "Attribute_1003"
              },
              {
                "name": "Size",
                "code": "6INCH",
                "isPriceImpacting": true,
                "businessObjectType": "Attribute",
                "id": "Attribute_1004",
                "attribute_id": "Attribute_1004"
              }
            ]
          }
        ]
      }
    ]
  },
  "pricingProcedureId": "9QMxx0000004CKKGA2",
  "configurationOverrides": {
    "skipWaterfall": true,
    "useSessionScopedContext": true,
    "persistContext": true,
    "referenceKey": "referenceKey-12345",
    "displayContext" : false,
    "taggedData": false
  }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`configurationOverrides`	Configuration Override Input	Parameters to override the pricing configuration.	Optional	60.0
`contextDefinitionId`	String	ID of the context definition that defines the structure of the input data.	Required	60.0
`contextMappingId`	String	ID of the context mapping that maps the input data to the context instance.	Required	60.0
`jsonDataString`	String	Data to hydrate the context, which must be in JSON format and passed as String. Pass the JSON	Required	60.0

Name	Type	Description	Required or Optional	Available Version
		data as String by using the stringify() method to convert the object to string. The keys in the `jsonDataString` property must be in accordance to the `contextMappingId` property sent in the request.		
`pricingProcedureId`	String	ID or API name of the pricing procedure used for calculating the prices. A pricing procedure is represented as an Expression Set Definition in the system.	Optional	60.0

Response body for POST

Pricing Output

API Execution Logs (GET)

Get the log details of a pricing API execution record by using the execution ID.

Resource

```
/connect/core-pricing/apiexecutionlogs/executionId
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/core-pricing/apiexecutionlogs/29646938297972
```

Available version

63.0

HTTP methods

GET

Path parameter for GET

Parameter Name	Type	Description	Required or Optional	Available Version
`executionId`	String	ID of the pricing process execution record.	Required	63.0

Response body for GET

Pricing Execution Waterfall Response

Pricing Process Execution (GET)

Get the execution details of a pricing process by using the execution ID.

Resource

```
/connect/core-pricing/pricing-process-execution/executionId
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/core-pricing/pricing-process-execution/29646938297972
```

Available version

63.0

HTTP methods

GET

Path parameter for GET

Parameter Name	Type	Description	Required or Optional	Available Version
`executionId`	String	ID of the pricing process execution record. The ID is generated each time a pricing process is executed.	Required	63.0

Query parameter for GET

Parameter Name	Type	Description	Required or Optional	Available Version
`executionType`	String	Type of execution that's defined internally within the pricing API. Valid values are: • `API_Execution` • `Discovery`—Discovery procedure • `Discovery_Line`—Discovery procedure for the line items. • `Pricing`—Pricing procedure • `Pricing_Line`—Pricing procedure for the line items. If the `executionType` parameter isn't specified, the API retrieves records for all the execution types that are associated with the specified execution ID.	Optional	63.0

Response body for GET

Pricing Process Execution Response

Pricing Process Execution for Line Items (GET)

Get the pricing execution details for the line items of a pricing process by using the execution ID and execution type.

Resource

```
/connect/core-pricing/pricing-process-execution/lineitems/executionId/executionType
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/core-pricing/pricing-process-execution/lineitems/29646938297972/Pricing_Line
```

Available version

63.0

HTTP methods

GET

Path parameters for GET

Parameter Name	Type	Description	Required or Optional	Available Version
`executionId`	String	ID of the pricing process execution record.	Required	63.0
`executionType`	String	Type of the execution that's defined internally within the pricing API. Valid values are: • `Pricing_Line` • `Discovery_Line`	Required	63.0

Response body for GET

Pricing Process Execution Details for Line Items

Pricing Data Sync (GET)

Sync pricing data to ensure that the lookup tables contain the latest pricing data.

To partially synchronize pricing data, use the Decision Table Refresh Action in a Flow. See Decision Table Refresh Action.

Resource

```
/connect/core-pricing/sync/pricingSyncOrigin
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/core-pricing/sync/syncData
```

Available version

60.0

HTTP methods

GET

Response body for GET

Pricing Generic Response

Pricing Recipe (GET)

Get the mapping details of pricing recipes to the associated pricing recipe table.

Resource

```
/connect/core-pricing/recipe
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/core-pricing/recipe
```

Available version

60.0

HTTP methods

GET

Response body for GET

Pricing Recipe Response

Pricing Recipe Mapping (POST)

Create a mapping between the pricing recipe and the Decision Tables. Post recipes with lookup tables or procedures.

Resource

```
/connect/core-pricing/recipe/mapping
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/core-pricing/recipe/mapping
```

Available version

60.0

HTTP methods

POST

Request body for POST

JSON example

```
{
    "recipeId" : "12Gxx0000005J9MEAU",
    "pricingRecipeLookUpTableInputRepresentations": [
    {
      lookupId: “12Gxx0000005J9MEAU”,
      pricingComponentType: “CustomDiscount”
  },
    {
      lookupId: “12Gxx0000005J9MEAU”,
      pricingComponentType: “CustomDiscount”
  }
 ],
    "pricingRecipeProcedureInputRepresentation" : {
    "procedureId" : "9QLxx0000004C92GAE"
```

```
}
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`pricingRecipeLookUpTableInputRepresentations`	Pricing Recipe LookUp Table Input[]	Input representation of the recipe mapping.	Required	60.0
`pricingRecipeProcedureInputRepresentation`	Pricing Recipe Procedure Input	Input representation of the procedure that's used in the pricing recipe.	Required	60.0
`recipeId`	String	ID of the pricing recipe.	Required	60.0

Response body for POST

Pricing Recipe Post

Pricing Versioned Revision Details (POST)

Create revisions of a pricing request with versions for adjustment entities.

Resource

```
/connect/core-pricing/versioned-revise-details
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/core-pricing/versioned-revise-details
```

Available version

60.0

HTTP methods

POST

Request body for POST

JSON example

This example shows the input for versioned revision details for attribute-based adjustment.

```
{
        "entityName":"AttributeBasedAdjustment",
        "id":"entityId",
        "priceAdjustmentId":"priceAdjustmentScheduleId",
        "productId":"ProductId",
        "productSellingModelId":"PsmId",
        "adjustmentType":"AdjustmentType",
```

```
        "adjustmentValue":"AdjustmentValue(Numeric)"",
        "effectiveFrom":"EffectiveFrom date",
        "effectiveTo":"EffectiveTo Date",
        "additionalFieldsToValueMap":{
        "attributeBasedAdjRuleId":"AttributeBasedAdjRuleId"
}
}
```

This example shows the input for versioned revision details for bundle-based adjustment.

```
 {
        "entityName": "BundleBasedAdjustment",
        "id": "entityId",
        "priceAdjustmentScheduleId": "priceAdjustmentScheduleId",
        "productId": "ProductId",
        "productSellingModelId": "PsmId",
        "adjustmentType": "AdjustmentType",
        "adjustmentValue": "AdjustmentValue(Numeric)",
        "effectiveFrom":"EffectiveFrom date",
        "effectiveTo":"EffectiveTo Date",
        "additionalFieldsToValueMap": {
          "rootBundleId": "RootBundleId",
          "parentProductId": "ParentProductId"
        }
 }
```

Properties

Name	Type	Description	Required or Optional	Available Version
`additionalFieldsToValueMap`	Map<String, String>	Map containing the additional fields specific to the entity.	Optional	60.0
`adjustmentType`	String	Adjustment type such as, percentage, amount, or override.	Required	60.0
`adjustmentValue`	String	Value for the adjustment.	Required	60.0
`effectiveFrom`	String	Date from when the adjustment is effective.	Required	60.0
`effectiveTo`	String	Date until when the adjustment is effective.	Optional	60.0
`entityName`	String	Name of the entity such as AttributeBasedAdjustment entity or BundleBasedAdjustment entity.	Required	60.0
`id`	String	ID of the record.	Required	60.0

Name	Type	Description	Required or Optional	Available Version
`priceAdjustmentScheduleId`	String	ID of the price adjustment schedule record.	Required	60.0
`productId`	String	Product ID of the record.	Required	60.0
`productSellingModelId`	String	Product selling model ID associated to the record.	Optional	60.0

Response body for POST

Pricing Versioned Revision Details

Pricing Waterfall (GET)

Get the persisted price waterfall that stores the process logs. Price waterfall provides insights into every step of the pricing process.

Resource

```
/connect/core-pricing/waterfall/lineItemId/executionId
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/core-pricing/waterfall/0bld/2yHbNNERCZr9jAe4gHS7?tagsToFilter=UnitPrice
```

Available version

60.0

HTTP methods

GET

Query parameters

Name	Type	Description	Required or Optional	Available Version
`tagsToFilter`	String	Comma-separated tags to filter.	Optional	61.0
`usageType`	String	Usage type of the waterfall log record. Valid values are: • `Pricing` • `Discovery` • `Rating`—Specifies that the record type is `Rating`. If this value is specified, the API creates a log of rating waterfall. See Rating Waterfall. The default value is `Pricing`.	Optional	62.0

Response body for GET

Line Item Waterfall Response

Pricing Waterfall (POST)

Create a log of price waterfall. Price waterfall provides insights into every step of the pricing process.

Resource

```
/connect/core-pricing/waterfall
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/core-pricing/waterfall
```

Available version

60.0

HTTP methods

POST

Request body for POST

JSON example

```
{
      "currencyCode": "USD",
      "executionEndTimestamp": "2023-07-31T20:11:29.625Z",
      "executionId": "executionId1",
      "executionStartTimestamp": null,
      "lineItemId": "item1",
      "output": {
          "Subtotal": 38.25,
          "ListPrice": 10,
          "NetUnitPrice": 7.65
  },
      "waterfall": [{
        "fieldToTagNameMapping": {
            "Product2Id": "ItemProduct",
            "Subtotal": "Subtotal",
            "Pricebook2Id": "Pricebook",
            "Quantity": "ItemQuantity",
            "LineItemId": "SalesTransactionSource",
            "ListPrice": "ItemListPrice"
  },
      "inputParameters": {
          "Product2Id": "01txx0000006i44AAA",
          "Pricebook2Id": "01sxx0000005q9xAAA",
          "Quantity": 5,
          "LineItemId": "item1"
    },
      "outputParameters": {
          "Subtotal": 50,
          "ListPrice": 10
      },
      "pricingElement": {
```

```
          "adjustments": [{
              "AdjustmentValue": "95.00",
              "AdjustmentType": "Amount"
           }],
          "description": null,
          "elementType": "ListPrice",
          "name": "List Price"
      },
          "sequence": 1
    },
    {
        "fieldToTagNameMapping": {
            "PriceAdjustmentScheduleId": "ItemDescription",
            "NetUnitPrice": "ItemNetUnitPrice",
            "Product2Id": "ItemProduct",
            "LowerBound": "ItemQuantity",
            "UpperBound": "ItemQuantity",
            "Subtotal": "Subtotal",
            "Quantity": "ItemQuantity",
            "LineItemId": "SalesTransactionSource",
            "InputUnitPrice": "ItemListPrice"
      },
        "inputParameters": {
            "PriceAdjustmentScheduleId": "84Xxx0000004CGSEA2",
            "Product2Id": "01txx0000006i44AAA",
            "LowerBound": 5,
            "UpperBound": 5,
            "Quantity": 5,
            "LineItemId": "item1",
            "InputUnitPrice": 10
       },
        "outputParameters": {
              "NetUnitPrice": 8.5,
              "Subtotal": 42.5
  },
        "pricingElement": {
              "adjustments": [{
              "AdjustmentValue": "15.00",
              "AdjustmentType": "Percentage"
  }],
      "description": null,
      "elementType": "VolumeDiscount",
      "name": "Volume Discount"
    },
      "sequence": 2
      }
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`contextDefinitionVersionId`	String	Context definition version ID of the pricing procedure.	Optional	60.0
`contextMappingId`	String	Context mapping ID of the pricing procedure.	Optional	60.0
`currencyCode`	String	Currency code such as, USD or INR.	Optional	60.0
`executionEndTimestamp`	String	End timestamp of procedure execution.	Optional	60.0
`executionId`	String	Execution ID for a particular execution of a pricing procedure.	Required	60.0
`executionStartTimestamp`	String	Start timestamp of procedure execution.	Optional	60.0
`lineItemId`	String	Line item ID for which the price is being calculated.	Required	60.0
`output`	Map<String, Object>	Output of the pricing procedure.	Optional	60.0
`waterfall`	Pricing Waterfall Input[]	Details of the pricing waterfall.	Required	60.0

Response body for POST

Pricing Generic Response

Procedure Plan Definitions (GET, POST)

Get the records of procedure plan definitions. Additionally, create a record of a procedure plan definition.

Resource

```
/connect/procedure-plan-definitions
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/
procedure-plan-definitions?isTemplate=true
```

Available version

62.0

HTTP methods

GET, POST

Request parameters for GET

Parameter Name	Type	Description	Required or Optional	Available Version
`isTemplate`	Boolean	Indicates whether to return a list of file-based definitions (`true`) or not (`false`). This API request returns a list of database-based definitions, by default.	Optional	62.0

Response body for GET

Procedure Plan Definitions

Request body for POST

JSON example

This example shows a sample request to create a procedure plan definition record by using the Procedure Plan Definitions (POST) API.

```
  {
  "description": "Definition for Quote",
  "developerName": "Quote_Definition_Sample",
  "name": "Quote_Definition_Sample",
  "primaryObject": "BusinessHours",
  "procedurePlanDefinitionVersions": [
    {
      "active": false,
      "contextDefinition": "SalesTransactionContext__stdctx",
      "readContextMapping": "QuoteEntitiesMapping",
      "saveContextMapping": "QuoteEntitiesMapping",
      "effectiveFrom": "2024-07-15T10:15:30.000Z",
      "developerName": "Quote_Definition_V1",
      "rank": 1
    }
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`description`	String	Description of the procedure plan definition.	Optional	62.0
`developerName`	String	Developer name of the procedure plan definition.	Required if you're invoking the Procedure Plan Definitions API (POST).	62.0
`name`	String	Name of the procedure plan definition.	Optional	62.0

Name	Type	Description	Required or Optional	Available Version
`primaryObject`	String	Source object that's used to create a procedure with rule-based criteria. This property value must be a valid object name and must be unique in the ProcedurePlanDefinition object.	Required if you're invoking the Procedure Plan Definitions API (POST) and if you're creating a procedure with rule-based criteria.	62.0
`procedurePlanDefinitionVersions`	Procedure Plan Definition Version Input[]	List of versions of a procedure plan definition.	Required	62.0
`recordId`	String	ID of the procedure plan definition record.	Required if you're invoking the Procedure Plan Definition By ID API (PATCH).	62.0

Response body for POST

Procedure Plan Generic

Procedure Plan Definition By ID (GET, PATCH, DELETE)

Get, update, or delete a procedure plan definition record by using the record ID.

Resource

```
/connect/procedure-plan-definitions/procedurePlanDefinitionId
```

The `procedurePlanDefinitionId` property value is the ID or name of the procedure plan definition record to perform the request for.

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/
procedure-plan-definitions/1FNxx0000004EsOGAU
```

Available version

62.0

HTTP methods

DELETE, GET, PATCH

You can delete a procedure plan definition only if it doesn't include any active procedure plan version.

Response body for GET

Procedure Plan Definition

Request body for PATCH

JSON example

This example shows a sample request to update a procedure plan definition by using the Procedure Plan Definition By ID (PATCH) API.

Note: The properties that aren't specified in the input are deleted when updating the record.

```
{
  "description": "Default definition patch update",
  "developerName": "Quote_Definition",
  "name": "Quote_Definition",
  "primaryObject": "Quote",
  "recordId": "1FNxx0000004EsOGAU"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
description	String	Description of the procedure plan definition.	Optional	62.0
developerName	String	Developer name of the procedure plan definition.	Required if you're invoking the Procedure Plan Definitions API (POST).	62.0
name	String	Name of the procedure plan definition.	Optional	62.0
primaryObject	String	Source object that's used to create a procedure with rule-based criteria. This property value must be a valid object name and must be unique in the ProcedurePlanDefinition object.	Required if you're invoking the Procedure Plan Definitions API (POST) and if you're creating a procedure with rule-based criteria.	62.0
procedurePlanDefinitionVersions	Procedure Plan Definition Version Input[]	List of versions of a procedure plan definition.	Required	62.0
recordId	String	ID of the procedure plan definition record.	Required if you're invoking the Procedure Plan Definition By ID API (PATCH).	62.0

Response body for PATCH

Procedure Plan Definition

Procedure Plan Evaluation By Object (POST)

Evaluate a procedure plan definition based on a primary object to check for prerequisites such as usage type and context mapping details.

Resource

```
/connect/procedure-plan-definitions/evaluate
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/
procedure-plan-definitions/evaluate
```

Available version

62.0

HTTP methods

POST

Request body for POST

JSON example

This example shows a sample request to evaluate a procedure plan definition by using a primary object.

```
{
  "idList": ["a01DU000000BylcYAC"],
  "evaluationDate": "2024-07-08T10:15:30.000Z",
  "sectionType": ["PricingProcedure"],
  "subSectionType": ["Revenue"]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`evaluationDate`	String	Date when the evaluation is applicable. This property value must be within the date range when the procedure plan definition is effective.	Required	62.0
`idList`	String[]	List of record IDs of the procedure plan definitions to be evaluated.	Required only if you're invoking the Procedure Plan Evaluation By Object (POST) API .	62.0
`sectionType`	String[]	Name of section to be evaluated. Valid values are: • `PricingProcedure` • `ProductDiscoveryProcedure` • `ProductQualificationProcedure` • `PricingDiscoveryProcedure` • `DiscountSpreadServiceProcedure`	Optional	62.0

Name	Type	Description	Required or Optional	Available Version
		• `RatingProcedure` • `Custom` • `RatingDiscoveryProcedure`		
`subSectionType`	String[]	Name of subsection to be evaluated.	Optional	62.0

The combination of the `sectionType` and `subSectionType` property values must be unique for every procedure plan version.

Response body for POST

Procedure Plan Evaluation Response

Procedure Plan Evaluation By Definition Name (POST)

Evaluate a procedure plan definition based on the name of a definition to check for prerequisites such as usage type and context mapping details.

Resource

```
/connect/procedure-plan-definitions/evaluate/procedurePlanDefinitionName
```

Resource example

```
https://yourInstance.salesforce.com/services/data
/v63.0/connect/procedure-plan-definitions/evaluate/Sample_Definition
```

Available version

62.0

HTTP methods

POST

Request body for POST

JSON example

This example shows a sample request to evaluate a procedure plan definition by using a definition name.

```
{
  "evaluationDate": "2024-07-08T10:15:30.000Z",
  "sectionType": ["PricingProcedure"],
  "subSectionType": ["Revenue"]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`evaluationDate`	String	Date when the evaluation is applicable. This property value must be within the	Required	62.0

Name	Type	Description	Required or Optional	Available Version
		date range when the procedure plan definition is effective.		
`idList`	String[]	List of record IDs of the procedure plan definitions to be evaluated.	Required only if you're invoking the Procedure Plan Evaluation By Object (POST) API .	62.0
`sectionType`	String[]	Name of section to be evaluated. Valid values are: • `PricingProcedure` • `ProductDiscoveryProcedure` • `ProductQualificationProcedure` • `PricingDiscoveryProcedure` • `DiscountSpreadServiceProcedure` • `RatingProcedure` • `Custom` • `RatingDiscoveryProcedure`	Optional	62.0
`subSectionType`	String[]	Name of subsection to be evaluated.	Optional	62.0

The combination of the `sectionType` and `subSectionType` property values must be unique for every procedure plan version.

Response body for POST

Procedure Plan Evaluation Response

Procedure Plan Version (POST)

Create records of a procedure plan version with details.

Resource

```
/connect/procedure-plan-definitions/procedurePlanDefinitionId/version
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/
procedure-plan-definitions/1FNxx0000004EsOGAU/version
```

Available version

62.0

HTTP methods

POST

Request body for POST

JSON example

```
{
  "active": false,
  "developerName": "sample_version_input",
  "effectiveFrom": "2024-07-09T00:00:00.000Z",
  "contextDefinition": "SalesTransactionContext__stdctx",
  "procedurePlanSections": [
    {
      "isInherited": false,
      "procedurePlanOptions": [
        {
          "saveContextMapping": "AssetToSalesTransactionMapping",
          "expressionSetDefinition": "9QAZ60000004ECOOA2",
          "expressionSetLabel": "Revenue_Default_Pricing_Procedure",
          "expressionSetApiName": "Revenue Default Pricing Procedure",
          "logic": "1 AND 2 AND 3",
          "priority": 1,
          "procedurePlanCriterion": [
            {
              "conditionSequence": 1,
              "fieldObject": "BillingCountry",
              "fieldPath": "BillingCountry",
              "literalValue": "test",
              "operator": "Equals",
              "dataType": "Text"
            },
            {
              "conditionSequence": 2,
              "fieldObject": "BillingPostalCode",
              "fieldPath": "BillingPostalCode",
              "literalValue": "sample",
              "operator": "Equals",
              "dataType": "Text"
            },
            {
              "conditionSequence": 3,
              "fieldObject": "LastActivityDate",
              "fieldPath": "LastActivityDate",
              "literalValue": "2024-07-14",
              "operator": "LessThan",
              "dataType": "Date"
            }
          ]
        }
      ],
      "resolutionType": "RuleBased",
      "sectionType": "PricingProcedure",
      "sequence": 1,
      "subSectionType": "PricingProcedure",
      "recordId": "1FRZ60000008OIAOA2"
    }
  ],
```

```
  "rank": 1,
  "readContextMapping": "ProductDiscoveryContextMapping",
  "saveContextMapping": "OrderEntitiesMapping"
}
```

Note: The properties that aren't specified in the input are deleted when updating the record.

Properties

Name	Type	Description	Required or Optional	Available Version
`active`	Boolean	Indicates whether this procedure plan definition version is active (`true`) or not (`false`). You can't edit or delete a procedure plan version that's in the active state.	Required	62.0
`contextDefinition`	String	Context definition that's associated with the procedure plan definition version record.	Required	62.0
`developerName`	String	Unique developer name of the procedure plan definition version.	Required	62.0
`effectiveFrom`	String	Date and time from when the procedure plan definition version comes into effect.	Required	62.0
`effectiveTo`	String	Date and time from when the procedure plan definition version is no longer in effect.	Required	62.0
`inheritedFrom`	String	Template this procedure plan definition version is created from.	This property is read-only.	62.0
`procedurePlanSections`	Procedure Plan Section Input[]	Procedure setup sections for a procedure plan definition. Each section enables the setup of a procedure type by using a rule-based criteria. Keep these considerations in mind when you modify this property. • You can edit or delete a procedure plan section if it isn't associated with an active procedure plan version. • You can create a procedure plan section with rule-based resolution type if the primary object isn't empty in the definition.	Required	62.0
`rank`	Integer	Current rank of the procedure plan definition version that's used to decide	Required	62.0

Name	Type	Description	Required or Optional	Available Version
		the sequence of execution of a procedure plan definition version.		
`readContextMapping`	String	Mapping that's used to read data from the mapped object and populate the context definition. This property value must be associated with a context definition.	Optional	62.0
`recordId`	String	ID of the procedure plan definition version record.	Required	62.0
`saveContextMapping`	String	Mapping that's used to save data from the context definition and populate the mapped object. This property value must be associated with a context definition.	Optional	62.0
`status`	String	Status of the procedure plan definition version record.	Optional	62.0

Response body for POST

Procedure Plan Generic

Procedure Plan Version Details (GET, PATCH, DELETE)

Get, update, or delete a procedure plan definition version record by using the record ID.

Resource

```
/connect/procedure-plan-definitions/versions/procedurePlanVersionId
```

The `procedurePlanVersionId` property value is the ID or name of the procedure plan version record to perform the request for.

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/
procedure-plan-definitions/versions/1Cvxx0000004E1ACAU
```

Available version

62.0

HTTP methods

DELETE, GET, PATCH

You can't delete a procedure plan version if it's the only procedure plan version in a procedure plan definition.

Response body for GET

Procedure Plan Definition Version

Request body for PATCH

JSON example

```
{
  "active": false,
  "developerName": "sample_version_input",
  "effectiveFrom": "2024-07-09T00:00:00.000Z",
  "contextDefinition": "SalesTransactionContext__stdctx",
  "procedurePlanSections": [
    {
      "isInherited": false,
      "procedurePlanOptions": [
        {
          "saveContextMapping": "AssetToSalesTransactionMapping",
          "expressionSetDefinition": "9QAZ60000004ECOOA2",
          "expressionSetLabel": "Revenue_Default_Pricing_Procedure",
          "expressionSetApiName": "Revenue Default Pricing Procedure",
          "logic": "1 AND 2 AND 3",
          "priority": 1,
          "procedurePlanCriterion": [
            {
              "conditionSequence": 1,
              "fieldObject": "BillingCountry",
              "fieldPath": "BillingCountry",
              "literalValue": "test",
              "operator": "Equals",
              "dataType": "Text"
            },
            {
              "conditionSequence": 2,
              "fieldObject": "BillingPostalCode",
              "fieldPath": "BillingPostalCode",
              "literalValue": "sample",
              "operator": "Equals",
              "dataType": "Text"
            },
            {
              "conditionSequence": 3,
              "fieldObject": "LastActivityDate",
              "fieldPath": "LastActivityDate",
              "literalValue": "2024-07-14",
              "operator": "LessThan",
              "dataType": "Date"
            }
          ]
        }
      ],
      "resolutionType": "RuleBased",
      "sectionType": "PricingProcedure",
      "sequence": 1,
      "subSectionType": "PricingProcedure",
      "recordId": "1FRZ60000008OIAOA2"
    }
  ],
```

```
  "rank": 1,
  "readContextMapping": "ProductDiscoveryContextMapping",
  "saveContextMapping": "OrderEntitiesMapping"
}
```

Note: The properties that aren't specified in the input are deleted when updating the record.

Properties

Name	Type	Description	Required or Optional	Available Version
`active`	Boolean	Indicates whether this procedure plan definition version is active (`true`) or not (`false`). You can't edit or delete a procedure plan version that's in the active state.	Required	62.0
`contextDefinition`	String	Context definition that's associated with the procedure plan definition version record.	Required	62.0
`developerName`	String	Unique developer name of the procedure plan definition version.	Required	62.0
`effectiveFrom`	String	Date and time from when the procedure plan definition version comes into effect.	Required	62.0
`effectiveTo`	String	Date and time from when the procedure plan definition version is no longer in effect.	Required	62.0
`inheritedFrom`	String	Template this procedure plan definition version is created from.	This property is read-only.	62.0
`procedurePlanSections`	Procedure Plan Section Input[]	Procedure setup sections for a procedure plan definition. Each section enables the setup of a procedure type by using a rule-based criteria. Keep these considerations in mind when you modify this property. • You can edit or delete a procedure plan section if it isn't associated with an active procedure plan version. • You can create a procedure plan section with rule-based resolution type if the primary object isn't empty in the definition.	Required	62.0
`rank`	Integer	Current rank of the procedure plan definition version that's used to decide	Required	62.0

Name	Type	Description	Required or Optional	Available Version
		the sequence of execution of a procedure plan definition version.		
`readContextMapping`	String	Mapping that's used to read data from the mapped object and populate the context definition. This property value must be associated with a context definition.	Optional	62.0
`recordId`	String	ID of the procedure plan definition version record.	Required	62.0
`saveContextMapping`	String	Mapping that's used to save data from the context definition and populate the mapped object. This property value must be associated with a context definition.	Optional	62.0
`status`	String	Status of the procedure plan definition version record.	Optional	62.0

Response body for PATCH

Procedure Plan Generic

Request Bodies

Learn more about the available Salesforce Pricing API request bodies.

Adjustment Details Input

Input representation of the adjustment details.

Configuration Override Input

Input representation of the details to override for a Pricing API configuration.

PBE Derived Pricing Input

Input representation of the request to get the source product for the Price Book Entry (PBE) derived pricing.

Pricing Input

Input representation of the details of a Pricing API request.

Pricing Recipe Input

Input representation to set up a pricing recipe page.

Pricing Recipe LookUp Table Input

Input representation of the lookup tables for the setup page recipe.

Pricing Recipe Procedure Input

Input representation of the procedure for the setup page recipe.

Pricing Request Input
Input representation of a pricing request.

Pricing Versioned Revision Details Input
Input representation of the versioned revision details.

Pricing Waterfall Input
Input representation of the pricing waterfall details.

Pricing Waterfall Log Input
Input representation of the request to create an explainability action log.

Procedure Plan Criterion Input
Input representation of the details of a procedure plan criterion.

Procedure Plan Definition Input
Input representation of the details of a procedure plan definition.

Procedure Plan Definition Version Input
Input representation of the details of a procedure plan definition version.

Procedure Plan Evaluation Input
Input representation of the details used to evaluate a procedure plan definition.

Procedure Plan Section Input
Input representation of the details of a procedure plan section.

Procedure Plan Option Input
Input representation of the details of a procedure plan option.

Adjustment Details Input

Input representation of the adjustment details.

JSON example

```
  "pricingElement": {
    "adjustments": [{
    "AdjustmentValue": "15.00",
    "AdjustmentType": "Percentage"
 }],
  "description": null,
  "elementType": "VolumeDiscount",
  "name": "Volume Discount"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
adjustments	Map<String, Object>[]	Details of the pricing element.	Optional	60.0
description	String	Description of the pricing element.	Optional	60.0

Name	Type	Description	Required or Optional	Available Version
elementType	String	Type of the pricing element.	Optional	60.0
name	String	Name of the pricing element.	Optional	60.0

Configuration Override Input

Input representation of the details to override for a Pricing API configuration.

JSON example

```
"configurationOverrides": {
     "skipWaterfall": true,
     "useSessionScopedContext": true,
     "persistContext": true,
     "referenceKey": "referenceKey-12345",
     "displayContext" : false,
     "taggedData": false
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
discoveryProcedure	String	Name of the discovery procedure to use to fetch the details of assets.	Optional	61.0
displayContext	Boolean	Indicates whether the context structure for pricing must be displayed (true) or not (false).	Optional	61.0
persistContext	Boolean	Indicates whether the context must be persisted as per the mapping (true) or not (false).	Optional	60.0
referenceKey	String	Reference ID that a consuming workstream provides in the API to search for specific logs in the Pricing Operations Console.	Optional	63.0
skipDiscovery	Boolean	Indicates whether the discovery procedure must be skipped (true) or not (false).	Optional	61.0
skipWaterfall	Boolean	Indicates whether the price waterfall must be skipped in the output response (true) or not (false).	Optional	60.0
taggedData	Boolean	Indicates whether the JSON data string can specify tags in the input instead of attributes (true) or not (false).	Optional	60.0

Name	Type	Description	Required or Optional	Available Version
`useSession ScopedContext`	Boolean	Indicates whether a session scoped context must be created (`true`) or request scoped context (`false`). The default is `false`.	Optional	60.0

PBE Derived Pricing Input

Input representation of the request to get the source product for the Price Book Entry (PBE) derived pricing.

JSON example

```
{
"productId":"01txx0000006i2SAAQ",
"pricebookEntryId":"01uxx0000008yYcAAI",
"effectiveFrom":"2020-01-01T22:53:20.000Z",
"effectiveTo":"2021-01-01T22:53:20.000Z"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`effectiveFrom`	String	Date from when the price book entry is effective.	Required	61.0
`effectiveTo`	String	Date until when the price book entry is effective.	Required	61.0
`pricebook EntryId`	String	ID of the price book entry.	Required	61.0
`productId`	String	ID of the price book.	Required	61.0

Pricing Input

Input representation of the details of a Pricing API request.

JSON example

```
{
  "contextDefinitionId": "11Oxx0000006PdxEAE",
  "contextMappingId": "11jxx0000004LDDAA2",
  "jsonDataString": {
    "Cart": [
      {
        "id": "cart_1001",
        "cart_id": "cart_1001",
        "PriceBookId": "PriceBookId_1001",
        "businessObjectType": "Cart",
        "CartItem": [
```

```
        {
          "id": "lineItem_1001",
          "line_item_id": "lineItem_1001",
          "Quantity": 7,
          "PriceType": "OneTime",
          "Frequency": "",
          "UOM": "",
          "businessObjectType": "CartItem",
          "product_id": "01txx0000006i44AAA",
          "UnitPrice": 6.8,
          "NetUnitPrice": 0,
          "Attribute": [
            {
              "name": "Color",
              "code": "RED",
              "isPriceImpacting": true,
              "businessObjectType": "Attribute",
              "id": "Attribute_1001",
              "attribute_id": "Attribute_1001"
            },
            {
              "name": "Size",
              "code": "10INCH",
              "isPriceImpacting": true,
              "businessObjectType": "Attribute",
              "id": "Attribute_1002",
              "attribute_id": "Attribute_1002"
            }
          ]
        },
        {
          "id": "lineItem_1002",
          "line_item_id": "lineItem_1002",
          "quantity": 3,
          "PriceType": "OneTime",
          "Frequency": "",
          "UOM": "",
          "businessObjectType": "CartItem",
          "product_id": "01txx0000006i2SAAQ",
          "unitprice": 6,
          "NetUnitPrice": 0,
          "Attribute": [
            {
              "name": "Color",
              "code": "BLUE",
              "isPriceImpacting": true,
              "businessObjectType": "Attribute",
              "id": "Attribute_1003",
              "attribute_id": "Attribute_1003"
            },
            {
              "name": "Size",
              "code": "6INCH",
              "isPriceImpacting": true,
```

```
                "businessObjectType": "Attribute",
                "id": "Attribute_1004",
                "attribute_id": "Attribute_1004"
              }
            ]
          }
        ]
      }
    ]
  },
  "pricingProcedureId": "9QMxx0000004CKKGA2",
  "configurationOverrides": {
    "skipWaterfall": true,
    "useSessionScopedContext": true,
    "persistContext": true,
    "referenceKey": "referenceKey-12345",
    "displayContext" : false,
    "taggedData": false
  }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`configurationOverrides`	Configuration Override Input	Parameters to override the pricing configuration.	Optional	60.0
`contextDefinitionId`	String	ID of the context definition that defines the structure of the input data.	Required	60.0
`contextMappingId`	String	ID of the context mapping that maps the input data to the context instance.	Required	60.0
`jsonDataString`	String	Data to hydrate the context, which must be in JSON format and passed as String. Pass the JSON data as String by using the stringify() method to convert the object to string. The keys in the `jsonDataString` property must be in accordance to the `contextMappingId` property sent in the request.	Required	60.0
`pricingProcedureId`	String	ID or API name of the pricing procedure used for calculating the prices. A pricing procedure is	Optional	60.0

Name	Type	Description	Required or Optional	Available Version
		represented as an Expression Set Definition in the system.		

Pricing Recipe Input

Input representation to set up a pricing recipe page.

JSON example

```
{
    "recipeId" : "12Gxx0000005J9MEAU",
    "pricingRecipeLookUpTableInputRepresentations": [
    {
      lookupId: "12Gxx0000005J9MEAU",
      pricingComponentType: "CustomDiscount"
  },
    {
      lookupId: "12Gxx0000005J9MEAU",
      pricingComponentType: "CustomDiscount"
  }
 ],
    "pricingRecipeProcedureInputRepresentation" : {
    "procedureId" : "9QLxx0000004C92GAE"
}
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`pricingRecipeLookUpTableInputRepresentations`	Pricing Recipe LookUp Table Input[]	Input representation of the recipe mapping.	Required	60.0
`pricingRecipeProcedureInputRepresentation`	Pricing Recipe Procedure Input	Input representation of the procedure that's used in the pricing recipe.	Required	60.0
`recipeId`	String	ID of the pricing recipe.	Required	60.0

Pricing Recipe LookUp Table Input

Input representation of the lookup tables for the setup page recipe.

JSON example

```
    "pricingRecipeLookUpTableInputRepresentations": [
    {
      lookupId: "12Gxx0000005J9MEAU",
      pricingComponentType: "CustomDiscount"
  },
    {
      lookupId: "12Gxx0000005J9MEAU",
      pricingComponentType: "CustomDiscount"
  }
 ]
```

Properties

Name	Type	Description	Required or Optional	Available Version
`lookupTableId`	String	ID of the decision table.	Optional	60.0
`pricingComponentType`	String	Pricing component types such as volume discount, custom discount, attribute-based discount, and bundle-based discount.	Optional	60.0

Pricing Recipe Procedure Input

Input representation of the procedure for the setup page recipe.

JSON example

```
    "pricingRecipeProcedureInputRepresentation" : {
    "procedureId" : "9QLxx0000004C92GAE"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`procedureId`	String	ID of the expression set.	Required	60.0

Pricing Request Input

Input representation of a pricing request.

JSON example

```
{
    "configurationOverrides": {
       "skipWaterfall": true,
       "useSessionScopedContext": true,
```

```
        "persistContext": true,
        "taggedData": false
    }
    "procedureName": "ES1"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
configurationOverrides	Configuration Override Input	Parameters to override pricing configuration.	Optional	60.0
procedureName	String	Name of the pricing procedure.	Optional	60.0

Pricing Versioned Revision Details Input

Input representation of the versioned revision details.

JSON example

This example shows the input for versioned revision details for attribute-based adjustment.

```
{
        "entityName":"AttributeBasedAdjustment",
        "id":"entityId",
        "priceAdjustmentId":"priceAdjustmentScheduleId",
        "productId":"ProductId",
        "productSellingModelId":"PsmId",
        "adjustmentType":"AdjustmentType",
        "adjustmentValue":"AdjustmentValue(Numeric)"",
        "effectiveFrom":"EffectiveFrom date",
        "effectiveTo":"EffectiveTo Date",
        "additionalFieldsToValueMap":{
        "attributeBasedAdjRuleId":"AttributeBasedAdjRuleId"
}
}
```

This example shows the input for versioned revision details for bundle-based adjustment.

```
 {
        "entityName": "BundleBasedAdjustment",
        "id": "entityId",
        "priceAdjustmentScheduleId": "priceAdjustmentScheduleId",
        "productId": "ProductId",
        "productSellingModelId": "PsmId",
        "adjustmentType": "AdjustmentType",
        "adjustmentValue": "AdjustmentValue(Numeric)",
        "effectiveFrom":"EffectiveFrom date",
        "effectiveTo":"EffectiveTo Date",
        "additionalFieldsToValueMap": {
          "rootBundleId": "RootBundleId",
```

```
          "parentProductId": "ParentProductId"
        }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
additionalFieldsToValueMap	Map<String, String>	Map containing the additional fields specific to the entity.	Optional	60.0
adjustmentType	String	Adjustment type such as, percentage, amount, or override.	Required	60.0
adjustmentValue	String	Value for the adjustment.	Required	60.0
effectiveFrom	String	Date from when the adjustment is effective.	Required	60.0
effectiveTo	String	Date until when the adjustment is effective.	Optional	60.0
entityName	String	Name of the entity such as AttributeBasedAdjustment entity or BundleBasedAdjustment entity.	Required	60.0
id	String	ID of the record.	Required	60.0
priceAdjustmentScheduleId	String	ID of the price adjustment schedule record.	Required	60.0
productId	String	Product ID of the record.	Required	60.0
productSellingModelId	String	Product selling model ID associated to the record.	Optional	60.0

Pricing Waterfall Input

Input representation of the pricing waterfall details.

JSON example

```
      "waterfall": [{
        "fieldToTagNameMapping": {
            "Product2Id": "ItemProduct",
            "Subtotal": "Subtotal",
            "Pricebook2Id": "Pricebook",
            "Quantity": "ItemQuantity",
```

```
            "LineItemId": "SalesTransactionSource",
            "ListPrice": "ItemListPrice"
  },
      "inputParameters": {
          "Product2Id": "01txx0000006i44AAA",
          "Pricebook2Id": "01sxx0000005q9xAAA",
          "Quantity": 5,
          "LineItemId": "item1"
    },
      "outputParameters": {
          "Subtotal": 50,
          "ListPrice": 10
      },
      "pricingElement": {
          "adjustments": [{
              "AdjustmentValue": "95.00",
              "AdjustmentType": "Amount"
           }],
          "description": null,
          "elementType": "ListPrice",
          "name": "List Price"
      },
          "sequence": 1
    },
    {
        "fieldToTagNameMapping": {
            "PriceAdjustmentScheduleId": "ItemDescription",
            "NetUnitPrice": "ItemNetUnitPrice",
            "Product2Id": "ItemProduct",
            "LowerBound": "ItemQuantity",
            "UpperBound": "ItemQuantity",
            "Subtotal": "Subtotal",
            "Quantity": "ItemQuantity",
            "LineItemId": "SalesTransactionSource",
            "InputUnitPrice": "ItemListPrice"
      },
        "inputParameters": {
            "PriceAdjustmentScheduleId": "84Xxx0000004CGSEA2",
            "Product2Id": "01txx0000006i44AAA",
            "LowerBound": 5,
            "UpperBound": 5,
            "Quantity": 5,
            "LineItemId": "item1",
            "InputUnitPrice": 10
       },
        "outputParameters": {
              "NetUnitPrice": 8.5,
              "Subtotal": 42.5
  },
        "pricingElement": {
              "adjustments": [{
              "AdjustmentValue": "15.00",
              "AdjustmentType": "Percentage"
  }],
```

```
      "description": null,
      "elementType": "VolumeDiscount",
      "name": "Volume Discount"
    },
      "sequence": 2
      }
  ]
```

Properties

Name	Type	Description	Required or Optional	Available Version
fieldToTagNameMapping	Map<String, String>	Mappings of field to tag names.	Optional	60.0
inputParameters	Map<String, Object>	Input parameters of the pricing element.	Optional	60.0
outputParameters	Map<String, Object>	Output parameters of the pricing element.	Optional	60.0
pricingElement	Adjustment Details Input	Details of the pricing element.	Optional	60.0
sequence	Integer	Sequence of the pricing element execution.	Optional	60.0

Pricing Waterfall Log Input

Input representation of the request to create an explainability action log.

JSON example

```
{
      "currencyCode": "USD",
      "executionEndTimestamp": "2023-07-31T20:11:29.625Z",
      "executionId": "executionId1",
      "executionStartTimestamp": null,
      "lineItemId": "item1",
      "output": {
          "Subtotal": 38.25,
          "ListPrice": 10,
          "NetUnitPrice": 7.65
  },
      "waterfall": [{
        "fieldToTagNameMapping": {
            "Product2Id": "ItemProduct",
            "Subtotal": "Subtotal",
            "Pricebook2Id": "Pricebook",
            "Quantity": "ItemQuantity",
            "LineItemId": "SalesTransactionSource",
```

```
            "ListPrice": "ItemListPrice"
  },
      "inputParameters": {
          "Product2Id": "01txx0000006i44AAA",
          "Pricebook2Id": "01sxx0000005q9xAAA",
          "Quantity": 5,
          "LineItemId": "item1"
    },
      "outputParameters": {
          "Subtotal": 50,
          "ListPrice": 10
      },
      "pricingElement": {
          "adjustments": [{
              "AdjustmentValue": "95.00",
              "AdjustmentType": "Amount"
           }],
          "description": null,
          "elementType": "ListPrice",
          "name": "List Price"
      },
          "sequence": 1
    },
    {
        "fieldToTagNameMapping": {
            "PriceAdjustmentScheduleId": "ItemDescription",
            "NetUnitPrice": "ItemNetUnitPrice",
            "Product2Id": "ItemProduct",
            "LowerBound": "ItemQuantity",
            "UpperBound": "ItemQuantity",
            "Subtotal": "Subtotal",
            "Quantity": "ItemQuantity",
            "LineItemId": "SalesTransactionSource",
            "InputUnitPrice": "ItemListPrice"
      },
        "inputParameters": {
            "PriceAdjustmentScheduleId": "84Xxx0000004CGSEA2",
            "Product2Id": "01txx0000006i44AAA",
            "LowerBound": 5,
            "UpperBound": 5,
            "Quantity": 5,
            "LineItemId": "item1",
            "InputUnitPrice": 10
       },
        "outputParameters": {
              "NetUnitPrice": 8.5,
              "Subtotal": 42.5
  },
        "pricingElement": {
              "adjustments": [{
              "AdjustmentValue": "15.00",
              "AdjustmentType": "Percentage"
  }],
      "description": null,
```

```
      "elementType": "VolumeDiscount",
      "name": "Volume Discount"
    },
      "sequence": 2
      }
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`contextDefinitionVersionId`	String	Context definition version ID of the pricing procedure.	Optional	60.0
`contextMappingId`	String	Context mapping ID of the pricing procedure.	Optional	60.0
`currencyCode`	String	Currency code such as, USD or INR.	Optional	60.0
`executionEndTimestamp`	String	End timestamp of procedure execution.	Optional	60.0
`executionId`	String	Execution ID for a particular execution of a pricing procedure.	Required	60.0
`executionStartTimestamp`	String	Start timestamp of procedure execution.	Optional	60.0
`lineItemId`	String	Line item ID for which the price is being calculated.	Required	60.0
`output`	Map<String, Object>	Output of the pricing procedure.	Optional	60.0
`waterfall`	Pricing Waterfall Input[]	Details of the pricing waterfall.	Required	60.0

Procedure Plan Criterion Input

Input representation of the details of a procedure plan criterion.

JSON example

```
        "procedurePlanCriterion": [
           {
             "conditionSequence": 1,
             "fieldObject": "BillingCountry",
             "fieldPath": "BillingCountry",
             "literalValue": "test",
             "operator": "Equals",
```

```
            "dataType": "Text"
          },
          {
            "conditionSequence": 2,
            "fieldObject": "BillingPostalCode",
            "fieldPath": "BillingPostalCode",
            "literalValue": "sample",
            "operator": "Equals",
            "dataType": "Text"
          },
          {
            "conditionSequence": 3,
            "fieldObject": "LastActivityDate",
            "fieldPath": "LastActivityDate",
            "literalValue": "2024-07-14",
            "operator": "LessThan",
            "dataType": "Date"
          }
        ]
```

Properties

Name	Type	Description	Required or Optional	Available Version
`conditionSequence`	Integer	Sequence to be followed to process the conditions defined in the procedure plan option. This property value must be unique within a procedure plan option.	Required	62.0
`dataType`	String	Data type of the field from the selected object.	Required	62.0
`fieldObject`	String	Value of the object field that's used to resolve the procedure plan option. This property value must belong to the primary object that's associated with the procedure plan definition, at a maximum two levels up in the hierarchy.	Required	62.0
`fieldPath`	String	Path of the field that's used in a procedure in relation to the object that the field belongs to. The field path must end with the object field that's associated with the procedure plan criterion.	Required	62.0
`literalValue`	String	User-defined value that's compared to the value of the sObject field value.	Optional	62.0

Name	Type	Description	Required or Optional	Available Version
operator	String	Operator that's used by the procedure plan criterion.	Required	62.0
recordId	String	ID of the procedure plan criterion record.	Required	62.0

Procedure Plan Definition Input

Input representation of the details of a procedure plan definition.

JSON example

This example shows a sample request to create a procedure plan definition record by using the Procedure Plan Definitions (POST) API.

```
  {
  "description": "Definition for Quote",
  "developerName": "Quote_Definition_Sample",
  "name": "Quote_Definition_Sample",
  "primaryObject": "BusinessHours",
  "procedurePlanDefinitionVersions": [
    {
      "active": false,
      "contextDefinition": "SalesTransactionContext__stdctx",
      "readContextMapping": "QuoteEntitiesMapping",
      "saveContextMapping": "QuoteEntitiesMapping",
      "effectiveFrom": "2024-07-15T10:15:30.000Z",
      "developerName": "Quote_Definition_V1",
      "rank": 1
    }
  ]
}
```

This example shows a sample request to update a procedure plan definition by using the Procedure Plan Definition By ID (PATCH) API.

Note: The properties that aren't specified in the input are deleted when updating the record.

```
{
  "description": "Default definition patch update",
  "developerName": "Quote_Definition",
  "name": "Quote_Definition",
  "primaryObject": "Quote",
  "recordId": "1FNxx0000004EsOGAU"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
description	String	Description of the procedure plan definition.	Optional	62.0

Name	Type	Description	Required or Optional	Available Version
developerName	String	Developer name of the procedure plan definition.	Required if you're invoking the Procedure Plan Definitions API (POST).	62.0
name	String	Name of the procedure plan definition.	Optional	62.0
primaryObject	String	Source object that's used to create a procedure with rule-based criteria. This property value must be a valid object name and must be unique in the ProcedurePlanDefinition object.	Required if you're invoking the Procedure Plan Definitions API (POST) and if you're creating a procedure with rule-based criteria.	62.0
procedurePlanDefinitionVersions	Procedure Plan Definition Version Input[]	List of versions of a procedure plan definition.	Required	62.0
recordId	String	ID of the procedure plan definition record.	Required if you're invoking the Procedure Plan Definition By ID API (PATCH).	62.0

Procedure Plan Definition Version Input

Input representation of the details of a procedure plan definition version.

JSON example

```
{
  "active": false,
  "developerName": "sample_version_input",
  "effectiveFrom": "2024-07-09T00:00:00.000Z",
  "contextDefinition": "SalesTransactionContext__stdctx",
  "procedurePlanSections": [
    {
      "isInherited": false,
      "procedurePlanOptions": [
        {
          "saveContextMapping": "AssetToSalesTransactionMapping",
          "expressionSetDefinition": "9QAZ60000004ECOOA2",
          "expressionSetLabel": "Revenue_Default_Pricing_Procedure",
          "expressionSetApiName": "Revenue Default Pricing Procedure",
          "logic": "1 AND 2 AND 3",
          "priority": 1,
          "procedurePlanCriterion": [
```

```
            {
              "conditionSequence": 1,
              "fieldObject": "BillingCountry",
              "fieldPath": "BillingCountry",
              "literalValue": "test",
              "operator": "Equals",
              "dataType": "Text"
            },
            {
              "conditionSequence": 2,
              "fieldObject": "BillingPostalCode",
              "fieldPath": "BillingPostalCode",
              "literalValue": "sample",
              "operator": "Equals",
              "dataType": "Text"
            },
            {
              "conditionSequence": 3,
              "fieldObject": "LastActivityDate",
              "fieldPath": "LastActivityDate",
              "literalValue": "2024-07-14",
              "operator": "LessThan",
              "dataType": "Date"
            }
          ]
        }
      ],
      "resolutionType": "RuleBased",
      "sectionType": "PricingProcedure",
      "sequence": 1,
      "subSectionType": "PricingProcedure",
      "recordId": "1FRZ60000008OIAOA2"
    }
  ],
  "rank": 1,
  "readContextMapping": "ProductDiscoveryContextMapping",
  "saveContextMapping": "OrderEntitiesMapping"
}
```

Note: The properties that aren't specified in the input are deleted when updating the record.

Properties

Name	Type	Description	Required or Optional	Available Version
`active`	Boolean	Indicates whether this procedure plan definition version is active (`true`) or not (`false`). You can't edit or delete a procedure plan version that's in the active state.	Required	62.0

Name	Type	Description	Required or Optional	Available Version
`contextDefinition`	String	Context definition that's associated with the procedure plan definition version record.	Required	62.0
`developerName`	String	Unique developer name of the procedure plan definition version.	Required	62.0
`effectiveFrom`	String	Date and time from when the procedure plan definition version comes into effect.	Required	62.0
`effectiveTo`	String	Date and time from when the procedure plan definition version is no longer in effect.	Required	62.0
`inheritedFrom`	String	Template this procedure plan definition version is created from.	This property is read-only.	62.0
`procedurePlanSections`	Procedure Plan Section Input[]	Procedure setup sections for a procedure plan definition. Each section enables the setup of a procedure type by using a rule-based criteria. Keep these considerations in mind when you modify this property. • You can edit or delete a procedure plan section if it isn't associated with an active procedure plan version. • You can create a procedure plan section with rule-based resolution type if the primary object isn't empty in the definition.	Required	62.0
`rank`	Integer	Current rank of the procedure plan definition version that's used to decide the sequence of execution of a procedure plan definition version.	Required	62.0
`readContextMapping`	String	Mapping that's used to read data from the mapped object and populate the context definition. This property value must be associated with a context definition.	Optional	62.0
`recordId`	String	ID of the procedure plan definition version record.	Required	62.0

Name	Type	Description	Required or Optional	Available Version
saveContextMapping	String	Mapping that's used to save data from the context definition and populate the mapped object. This property value must be associated with a context definition.	Optional	62.0
status	String	Status of the procedure plan definition version record.	Optional	62.0

Procedure Plan Evaluation Input

Input representation of the details used to evaluate a procedure plan definition.

JSON example

This example shows a sample request to evaluate a procedure plan definition by using a primary object.

```
{
  "idList": ["a01DU000000BylcYAC"],
  "evaluationDate": "2024-07-08T10:15:30.000Z",
  "sectionType": ["PricingProcedure"],
  "subSectionType": ["Revenue"]
}
```

This example shows a sample request to evaluate a procedure plan definition by using a definition name.

```
{
  "evaluationDate": "2024-07-08T10:15:30.000Z",
  "sectionType": ["PricingProcedure"],
  "subSectionType": ["Revenue"]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
evaluationDate	String	Date when the evaluation is applicable. This property value must be within the date range when the procedure plan definition is effective.	Required	62.0
idList	String[]	List of record IDs of the procedure plan definitions to be evaluated.	Required only if you're invoking the Procedure Plan Evaluation By Object (POST) API .	62.0

Name	Type	Description	Required or Optional	Available Version
`sectionType`	String[]	Name of section to be evaluated. Valid values are: • `PricingProcedure` • `ProductDiscoveryProcedure` • `ProductQualificationProcedure` • `PricingDiscoveryProcedure` • `DiscountSpreadServiceProcedure` • `RatingProcedure` • `Custom` • `RatingDiscoveryProcedure`	Optional	62.0
`subSectionType`	String[]	Name of subsection to be evaluated.	Optional	62.0

The combination of the `sectionType` and `subSectionType` property values must be unique for every procedure plan version.

Procedure Plan Section Input

Input representation of the details of a procedure plan section.

JSON example

```
"procedurePlanSections": [
  {
    "isInherited": false,
    "procedurePlanOptions": [
      {
        "saveContextMapping": "AssetToSalesTransactionMapping",
        "expressionSetDefinition": "9QAZ60000004ECOOA2",
        "expressionSetLabel": "Revenue_Default_Pricing_Procedure",
        "expressionSetApiName": "Revenue Default Pricing Procedure",
        "logic": "1 AND 2 AND 3",
        "priority": 1,
        "procedurePlanCriterion": [
          {
            "conditionSequence": 1,
            "fieldObject": "BillingCountry",
            "fieldPath": "BillingCountry",
            "literalValue": "test",
            "operator": "Equals",
            "dataType": "Text"
          },
          {
            "conditionSequence": 2,
            "fieldObject": "BillingPostalCode",
            "fieldPath": "BillingPostalCode",
            "literalValue": "sample",
```

```
              "operator": "Equals",
              "dataType": "Text"
            },
            {
              "conditionSequence": 3,
              "fieldObject": "LastActivityDate",
              "fieldPath": "LastActivityDate",
              "literalValue": "2024-07-14",
              "operator": "LessThan",
              "dataType": "Date"
            }
          ]
        }
      ],
      "resolutionType": "RuleBased",
      "sectionType": "PricingProcedure",
      "sequence": 1,
      "subSectionType": "PricingProcedure",
      "recordId": "1FRZ60000008OIAOA2"
    }
  ]
```

Properties

Name	Type	Description	Required or Optional	Available Version
`isInherited`	Boolean	Indicates whether the procedure plan section is inherited from a template (`true`) or not (`false`).	This property is read-only.	62.0
`procedurePlanOptions`	Procedure Plan Option Input[]	List of procedure plan options that defines a group of criteria. You can edit or delete a procedure plan option only if it isn't associated with an active procedure plan version.	Required	62.0
`recordId`	String	ID of the procedure plan section record.	Required	62.0
`resolutionType`	String	Type of resolution used to filter the procedure. You can't edit this property value if the procedure plan section includes a procedure plan option record.	Required	62.0
`sectionType`	String	Type of section. Valid values are: • `PricingProcedure` • `ProductDiscoveryProcedure` • `ProductQualificationProcedure` • `PricingDiscoveryProcedure` • `DiscountSpreadServiceProcedure` • `RatingProcedure`	Required	62.0

Name	Type	Description	Required or Optional	Available Version
		• `Custom` • `RatingDiscoveryProcedure`		
`sequence`	Integer	Sequence to be followed for the processing of the procedures. This property value must be greater than 0 and must be unique for a procedure plan section associated with a procedure plan version.	Required	62.0
`subSectionType`	String	Procedure subsection added to the procedure plan definition.	Required	62.0

Procedure Plan Option Input

Input representation of the details of a procedure plan option.

JSON example

```
      "procedurePlanOptions": [
        {
          "saveContextMapping": "AssetToSalesTransactionMapping",
          "expressionSetDefinition": "9QAZ60000004ECOOA2",
          "expressionSetLabel": "Revenute_Default_Pricing_Procedure",
          "expressionSetApiName": "Revenue Default Pricing Procedure",
          "logic": "1 AND 2 AND 3",
          "priority": 1,
          "procedurePlanCriterion": [
            {
              "conditionSequence": 1,
              "fieldObject": "BillingCountry",
              "fieldPath": "BillingCountry",
              "literalValue": "test",
              "operator": "Equals",
              "dataType": "Text"
            },
            {
              "conditionSequence": 2,
              "fieldObject": "BillingPostalCode",
              "fieldPath": "BillingPostalCode",
              "literalValue": "sample",
              "operator": "Equals",
              "dataType": "Text"
            },
            {
              "conditionSequence": 3,
              "fieldObject": "LastActivityDate",
              "fieldPath": "LastActivityDate",
              "literalValue": "2024-07-14",
              "operator": "LessThan",
              "dataType": "Date"
```

```
            }
          ]
        }
      ]
```

Properties

Name	Type	Description	Required or Optional	Available Version
`expressionSetApiName`	String	API name of the expression set.	Optional	62.0
`expressionSetDefinition`	String	Expression set definition that's associated with this procedure plan option record.	Required	62.0
`expressionSetLabel`	String	Label of the expression set that's associated with this procedure plan option record.	Optional	62.0
`logic`	String	Computation logic for the conditions applied to a procedure plan option. This property value must be blank if the resolution type is default.	Optional	62.0
`priority`	Integer	Priority for the specified criteria. This property value must be greater than 0 and must be unique within a procedure plan section.	Required	62.0
`procedurePlanCriterion`	Procedure Plan Criterion Input[]	Details of the rule-based criteria for the procedure. You can edit or delete a procedure plan criterion only if it isn't associated with an active procedure plan version.	Optional	62.0
`readContextMapping`	String	Mapping that's used to read from the mapped object and populate the context definition. This property value must be associated with a context definition.	Optional	62.0
`recordId`	String	ID of the procedure plan option record.	Required	62.0
`saveContextMapping`	String	Mapping that's used to save data from the context definition and populate the mapped object. This property value must be associated with a context definition.	Optional	62.0

Response Bodies

Learn more about the available Salesforce Pricing API response bodies.

Adjustment Details
Output representation of a pricing adjustment request.

API Execution Log Response
Output representation of the execution log of a pricing waterfall request.

Line Item Waterfall Response
Output representation of the line item waterfall response.

PBE Derived Pricing
Output representation of the response that includes the source product for the Price Book Entry (PBE) derived pricing.

Pricing Error Response
Output representation of the pricing error response.

Pricing Execution Waterfall Response
Output representation of the execution process that's associated with a pricing waterfall.

Pricing Generic Response
Output representation of a pricing data sync request.

Pricing Output
Output representation of a Salesforce pricing request.

Pricing Recipe LookUp Table Response
Output representation of a pricing recipe lookup table.

Pricing Recipe
Output representation of the pricing recipe information table.

Pricing Recipe Post
Output representation of the pricing recipe after the API request.

Pricing Recipe Response
Output representation of the pricing recipe.

Pricing Response
Output representation of the pricing request.

Pricing Result
Output representation of the pricing result.

Pricing Result Error
Output representation of the pricing result error.

Pricing Versioned Revision Details
Output representation of the versioned revision details.

Pricing Waterfall Response
Output representation of a pricing waterfall request.

Procedure Plan Criterion
Output representation of the details of a procedure plan criterion.

Procedure Plan Definition

Output representation of the details of a single procedure plan definition.

Procedure Plan Definition Version

Output representation of the version details of a procedure plan definition.

Procedure Plan Definitions

Output representation of the details of procedure plan definitions.

Procedure Plan Generic

Output representation of the details of the created procedure plan definition record.

Procedure Plan Generic Error

Output representation of the error details related to the procedure plan definitions.

Procedure Plan Option

Output representation of the details of a procedure plan option.

Procedure Plan Section

Output representation of the details of a procedure plan section.

Procedure Plan Section Evaluation Runtime

Output representation of the results from the procedure plan evaluation.

Procedure Plan Evaluation

Output representation of the evaluation details of a procedure plan definition.

Procedure Plan Evaluation Response

Output representation of the evaluation details of a procedure plan definition.

Procedure Plan Evaluation Result

Output representation of the evaluation result of a procedure plan definition.

Pricing Process Execution Details for Line Items

Output representation of the pricing process execution details for the line items along with the error details and response generation status.

Line Item Details Response

Output representation of the pricing process execution details for the line items.

Pricing Process Execution Response

Output representation of the details of a pricing process execution.

Pricing Process Execution List

Output representation of the execution details for different types of the pricing processes.

Adjustment Details

Output representation of a pricing adjustment request.

JSON example

```
"pricingElement": {
     "adjustments": [{
     "adjustmentType": null,
     "adjustmentValue": null
   }],
```

```
      "name": "List Price",
      "elementType": "ListPrice"
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
adjustments	Map<String, Object>[]	Details of the pricing element.	Small, 60.0	60.0
description	String	Description of the pricing element.	Small, 60.0	60.0
elementType	String	Type of the pricing element.	Small, 60.0	60.0
name	String	Name of the pricing element.	Small, 60.0	60.0

API Execution Log Response

Output representation of the execution log of a pricing waterfall request.

JSON example

```
{
  "message": {The Pricing API execution was successful.},
  "pricingElement": {
    "adjustments": [
      {
        "adjustmentType": null,
        "adjustmentValue": null
      }
    ],
    "name": "List Price",
    "elementType": "ListPrice"
  }
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
message	String []	Message of the API execution.	Small, 63.0	63.0
pricingElement	Adjustment Details	Details of the price adjustment of a pricing element.	Small, 63.0	63.0

Line Item Waterfall Response

Output representation of the line item waterfall response.

JSON example

```
{
       "currencyCode": "USD",
       "error": null,
```

```
        "executionEndTimestamp": "2023-07-31T20:11:29.625Z",
        "executionId": "gdLVwn2x1uats2xWMAjV",
        "executionStartTimestamp": null,
        "lineItemId": "item1",
        "success": true,
        "usageType":"Pricing",
        "output": {
           "quantity": "10",
           "netUnitPrice": "10",
           "subtotal": "100"
         },
         "waterfall": []
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`contextDefinitionVersionId`	String	Context definition version ID of the pricing procedure.	Small, 60.0	60.0
`contextMappingId`	String	Context mapping ID of the record.	Small, 60.0	60.0
`currencyCode`	String	Currency code. For example, USD or INR.	Small, 60.0	60.0
`error`	Pricing Error Response	Details of any errors.	Small, 60.0	60.0
`executionEndTimestamp`	String	End timestamp of procedure execution.	Small, 60.0	60.0
`executionId`	String	Execution ID of a particular execution of a pricing procedure.	Small, 60.0	60.0
`executionStartTimestamp`	String	Start timestamp of procedure execution.	Small, 60.0	60.0
`lineItemId`	String	Line item ID for which the price is being calculated.	Small, 60.0	60.0
`output`	Map<String, Object>	Output of the pricing procedure.	Small, 60.0	60.0
`success`	Boolean	Indicates whether the API request is successful (`true`) or not (`false`).	Small, 60.0	60.0
`usageType`	String	Usage type of the waterfall log record.	Small, 62.0	62.0
`waterfall`	Pricing Waterfall Response[]	Details of the price waterfall.	Small, 60.0	60.0

PBE Derived Pricing

Output representation of the response that includes the source product for the Price Book Entry (PBE) derived pricing.

JSON example

```
{
"productId":"01txx0000006i2SAAQ",
"pricebookEntryId":"01uxx0000008yYcAAI",
"effectiveFrom":"2020-01-01T22:53:20.000Z",
"effectiveTo":"2021-01-01T22:53:20.000Z"
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`error`	Pricing Error Response[]	Displays the error while processing the request.	Small, 61.0	61.0
`isSuccess`	Boolean	Indicates whether the request is successful (`true`) or not (`false`).	Small, 61.0	61.0
`source ProceductId`	String	ID of the source product.	Small, 61.0	61.0

Pricing Error Response

Output representation of the pricing error response.

Property Name	Type	Description	Filter Group and Version	Available Version
`errorCode`	String	Indicates the error code.	Small, 60.0	60.0
`message`	String	Specifies the message stating the reason for the error, if any.	Small, 60.0	60.0

Pricing Execution Waterfall Response

Output representation of the execution process that's associated with a pricing waterfall.

JSON example

```
{
  "apiExecutionId": "263369316770986",
  "apiExecutionLogRepresentationList": [
    {
      "message": [
       "The Pricing API couldn't be run. Try again, and if the issue persists, ask your
 admin for help."
      ]
    }
  ],
  "executionId": "263369316895959",
```

```
  "referenceKey": "referenceKey-ABCD",
  "success": false,
  "usageType": "Api_Execution"
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
apiExecutionId	String	Unique execution ID that's generated each time a pricing API is executed.	Small, 63.0	Small, 63.0
apiExecutionLogRepresentationList	API Execution Log Response[]	List of API execution logs.	Small, 63.0	63.0
error	Pricing Error Response	Error details of the pricing execution process.	Small, 63.0	Small, 63.0
executionId	String	Unique ID that's generated each time a pricing process is executed.	Small, 63.0	Small, 63.0
referenceKey	String	The reference ID that a consuming workstream provides in the API to search for the specific logs in the Pricing Operations Console.	Small, 63.0	Small, 63.0
success	Boolean	Indicates whether the API execution is successful (true) or not (false).	Small, 63.0	Small, 63.0
usageType	String	Usage type of the API execution.	Small, 63.0	Small, 63.0

Pricing Generic Response

Output representation of a pricing data sync request.

JSON example

```
{
  "success": true
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
error	Pricing Error Response	Details from the pricing error response.	Small, 60.0	60.0
success	Boolean	Indicates whether the request is successful (true) or not (false).	Small, 60.0	60.0

Pricing Output

Output representation of a Salesforce pricing request.

JSON example

```
{
"apiExecutionId": "612228038743152",
"pricingExecutionId": "612229738898095",
"pricingResult": {
"subtotal": [
{
"dataPath": [
"cart_1001",
"lineItem_1002"
],
"value": 300.0,
"errors": [],
"isSuccess": true
},
{
"dataPath": [
"cart_1001",
"lineItem_1001"
],
"value":400.0,
"errors": [],
"isSuccess": true
}
],
"netunitprice": [
{
"dataPath": [
"cart_1001",
"lineItem_1002"
],
"value": xx,
"errors": [],
"isSuccess": true
},
{
"dataPath": [
"cart_1001",
"lineItem_1001"
],
"value": xx,
"errors": [],
"isSuccess": true
}
]
},
"pricingResultErrors": [],
"status": "Completed",
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
apiExecutionId	String	Unique ID that's generated each time a pricing API is executed.	Small, 63.0	63.0
error	Pricing Error Response	Displays the error encountered when the request is processed. For example, a pricing procedure isn't found.	Small, 60.0	60.0
pricingExecutionId	String	Unique ID that's generated each time a pricing process is executed.	Small, 63.0	63.0
pricingResult	Pricing Result	Represents the outcomes associated with the output tags defined in the contextual definition for which the pricing engine establishes values. The initial attribute name is substituted for the output tag's designation. For instance, if the original attribute name specified in the Context Definition is "Subtotal," but during contextual setup, the output tag is denoted as "Total Price," the API output exhibits the initial attribute name "Subtotal" in the response.	Small, 60.0	60.0
pricingResultErrors	Pricing Result Error[]	Errors from the pricing request, if any.	Small, 60.0	60.0
status	String	Status of the pricing request. Valid values are: • Completed — Pricing is completed for all the line items. • Partially Completed — Pricing is completed for some line items. • Failed — Pricing isn't completed for the line items.	Small, 60.0	60.0

Pricing Recipe LookUp Table Response

Output representation of a pricing recipe lookup table.

JSON example

```
"decisionTables": [
  {
    "id": "0lDxx00000000T3EAI",
    "isInternal": true,
    "pricingComponentType": "ListPrice"
  },
```

```
        {
          "id": "0lDxx00000000T4EAI",
          "isInternal": true,
          "pricingComponentType": "VolumeDiscount"
        },
        {
          "id": "0lDxx00000000HlEAI",
          "isInternal": false,
          "pricingComponentType": "CustomDiscount"
        }
      ]
```

Property Name	Type	Description	Filter Group and Version	Available Version
`id`	String	ID of the pricing recipe table mapping.	Small, 60.0	60.0
`isInternal`	Boolean	Indicates if the decision table is available (`true`) or not (`false`).	Small, 60.0	60.0
`pricingComponentType`	String	Price component types such as, custom discount, volume discount, attribute-based discount, bundle-based discount, and list price.	Small, 60.0	60.0

Pricing Recipe

Output representation of the pricing recipe information table.

JSON example

```
"recipes": [
  {
    "active": false,
    "createdBy": "autoproc@00dxx0000006gmjea2",
    "createdOn": "2023-07-15T13:12:38.000Z",
    "decisionTables": [
      {
        "id": "0lDxx00000000T3EAI",
        "isInternal": true,
        "pricingComponentType": "ListPrice"
      },
      {
        "id": "0lDxx00000000T4EAI",
        "isInternal": true,
        "pricingComponentType": "VolumeDiscount"
      },
      {
        "id": "0lDxx00000000HlEAI",
        "isInternal": false,
        "pricingComponentType": "CustomDiscount"
      }
    ],
```

```
      "developerName": "NGPDefaultRecipe",
      "id": "12Gxx0000005Ka4EAE",
      "name": "NGPDefaultRecipe",
      "procedureCreatedBy": "",
      "procedureCreatedOn": "2023-09-19T11:39:18.983Z",
      "procedureId": "",
      "procedureName": ""
    }]
```

Property Name	Type	Description	Filter Group and Version	Available Version
`active`	Boolean	Indicates whether the recipe is active (`true`) or not (`false`).	Small, 60.0	60.0
`createdBy`	String	Details on who created the recipe.	Small, 60.0	60.0
`createdOn`	String	Date when the recipe was created.	Small, 60.0	60.0
`decisionTables`	Pricing Recipe LookUp Table Response []	Decision tables linked to the recipe.	Small, 60.0	60.0
`developerName`	String	API name of the recipe.	Small, 60.0	60.0
`id`	String	ID of the recipe.	Small, 60.0	60.0
`name`	String	Name of the recipe.	Small, 60.0	60.0
`procedureCreatedBy`	String	Details on who created the procedure.	Small, 60.0	60.0
`procedureCreatedOn`	String	Date when the procedure was created.	Small, 60.0	60.0
`procedureId`	String	ID of the procedure.	Small, 60.0	60.0
`procedureName`	String	Name of the procedure.	Small, 60.0	60.0

Pricing Recipe Post

Output representation of the pricing recipe after the API request.

JSON example

```
{
isSuccess : true
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`error`	Pricing Error Response	Details from the pricing error response.	Small, 60.0	60.0

Property Name	Type	Description	Filter Group and Version	Available Version
isSuccess	Boolean	Indicates whether the response was calculated successfully (true) or not (false).	Small, 60.0	60.0

Pricing Recipe Response

Output representation of the pricing recipe.

JSON example

```
{
  "recipes": [
    {
      "active": false,
      "createdBy": "autoproc@00dxx0000006gmjea2",
      "createdOn": "2023-07-15T13:12:38.000Z",
      "decisionTables": [
        {
          "id": "0lDxx00000000T3EAI",
          "isInternal": true,
          "pricingComponentType": "ListPrice"
        },
        {
          "id": "0lDxx00000000T4EAI",
          "isInternal": true,
          "pricingComponentType": "VolumeDiscount"
        },
        {
          "id": "0lDxx00000000HlEAI",
          "isInternal": false,
          "pricingComponentType": "CustomDiscount"
        }
      ],
      "developerName": "NGPDefaultRecipe",
      "id": "12Gxx0000005Ka4EAE",
      "name": "NGPDefaultRecipe",
      "procedureCreatedBy": "",
      "procedureCreatedOn": "2023-09-19T11:39:18.983Z",
      "procedureId": "",
      "procedureName": ""
    }],
  "success": true
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
recipes	Pricing Recipe Output Representation[]	Representation of the pricing recipe.	Small, 60.0	60.0

Property Name	Type	Description	Filter Group and Version	Available Version
`success`	Boolean	Indicates if the request is successful (`true`) or not (`false`).	Small, 60.0	60.0

Pricing Response

Output representation of the pricing request.

JSON example

```
{
    "success": true,
    "executionId": "zu81o5hBCrFzyd5LWZk"
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`error`	Pricing Error Response	Errors while processing the request, if any.	Small, 60.0	60.0
`executionId`	String	Auto-generated alphanumeric string for correlation to extract async waterfall and context persistence status.	Small, 60.0	60.0
`success`	Boolean	Indicates if the request is successful (`true`) or not (`false`).	Small, 60.0	60.0

Pricing Result

Output representation of the pricing result.

JSON example

```
"pricingResult": {
"subtotal": [
{
"dataPath": [
"cart_1001",
"lineItem_1002"
],
"value": 300.0,
"errors": [],
"isSuccess": true
},
{
"dataPath": [
"cart_1001",
"lineItem_1001"
],
```

```
"value":400.0,
"errors": [],
"isSuccess": true
}
],
"netunitprice": [
{
"dataPath": [
"cart_1001",
"lineItem_1002"
],
"value": xx,
"errors": [],
"isSuccess": true
},
{
"dataPath": [
"cart_1001",
"lineItem_1001"
],
"value": xx,
"errors": [],
"isSuccess": true
}
]
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`dataPath`	String	Includes the entire data route for the specific element starting from the root node. The request must include the ID to construct the accurate data route. For example, if a `jsonDataString` property comprises a Cart [Id = Cart1] and its associated Cart Item [Id = CartItem1], then the data route for CartItem appears as [Cart1, CartItem1].	Small, 60.0	60.0
`errors`	Pricing Error Response[]	Displays processing errors related to the element as recognized by the data path.	Small, 60.0	60.0
`isSuccess`	Boolean	Displays if processing of the element for the specified data path is successful or not.	Small, 60.0	60.0
`value`	Object	Displays the value of the element into consideration. Element is uniquely identify by the data path.	Small, 60.0	60.0

Pricing Result Error

Output representation of the pricing result error.

JSON example

```
"pricingResultErrors": {
"Aggregateprice": [
{
"dataPath": [
"cart_1001",
],


"errors": [
{
"errorCode": "Dummy"
"message":
 }
       ]
   }
 ]
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`dataPath`	String[]	Includes the entire data route for the specific element starting from the root node. The request must include the ID to construct the accurate data route. For example, if a `jsonDataString` property comprises a Cart [Id = Cart1] and its associated Cart Item [Id = CartItem1], then the data route for CartItem appears as [Cart1, CartItem1].	Small, 60.0	60.0
`errors`	Pricing Error Response	Displays processing errors related to the element as recognized by the data path.	Small, 60.0	60.0

Pricing Versioned Revision Details

Output representation of the versioned revision details.

Property Name	Type	Description	Filter Group and Version	Available Version
`error`	Pricing Error Response	Details from the pricing error response.	Small, 60.0	60.0
`success`	Boolean	Indicates whether the request is successful (`true`) or not (`false`).	Small, 60.0	60.0

Pricing Waterfall Response

Output representation of a pricing waterfall request.

JSON example

```
{
  "inputParameters": {
    "productId": "01txx0000006i2SAAQ",
    "pricebookId": "01sxx0000005ptpAAA",
    "pricingModelType": "OneTime"
  },
  "fieldToTagNameMapping": {
    "Product2Id": "ItemProduct",
    "Subtotal": "Subtotal",
    "Pricebook2Id": "Pricebook",
    "Quantity": "ItemQuantity",
    "LineItemId": "SalesTransactionSource",
    "ListPrice": "ItemListPrice"
  },
  "sequence": 0,
  "outputParameters": {
    "listPrice": "10"
  },
  "pricingElement": {
    "adjustments": [
      {
        "adjustmentType": null,
        "adjustmentValue": null
      }
    ],
    "name": "List Price"
  }
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`fieldToTagNameMapping`	Map<String, String>	Mappings of field to tag names.	Small, 60.0	60.0
`inputParameters`	Map<String, Object>	Parameters of pricing element input.	Small, 60.0	60.0
`outputParameters`	Map<String, Object>	Parameters of pricing element output.	Small, 60.0	60.0
`pricingElement`	Adjustment Details	Details of the price adjustment of a pricing element.	Small, 60.0	60.0
`sequence`	Integer	Sequence of pricing element execution.	Small, 60.0	60.0

Procedure Plan Criterion

Output representation of the details of a procedure plan criterion.

JSON example

```
"procedurePlanCriterion": [
        {
          "conditionSequence": 1,
          "dataType": "Text",
          "fieldObject": "BillingCountry",
          "fieldPath": "BillingCountry",
          "isSuccess": true,
          "literalValue": "test",
          "operator": "Equals",
          "recordId": "1FiZ60000004C9cKAE"
         },
         {
          "conditionSequence": 2,
          "dataType": "Text",
          "fieldObject": "BillingPostalCode",
          "fieldPath": "BillingPostalCode",
          "isSuccess": true,
          "literalValue": "pramit",
          "operator": "Equals",
          "recordId": "1FiZ60000004C9dKAE"
          },
          {
            "conditionSequence": 3,
            "dataType": "Date",
            "fieldObject": "LastActivityDate",
            "fieldPath": "LastActivityDate",
            "isSuccess": true,
            "literalValue": "2024-07-14",
            "operator": "LessThan",
            "recordId": "1FiZ60000004C9eKAE"
           }
         ]
```

Property Name	Type	Description	Filter Group and Version	Available Version
`conditionSequence`	Integer	Sequence to be followed to process the conditions defined in the procedure plan option.	Small, 62.0	62.0
`dataType`	String	Data type of the field from the selected object.	Small, 62.0	62.0
`error`	Procedure Plan Generic Error[]	Details of the error encountered during the processing of the API request.	Small, 62.0	62.0
`fieldObject`	String	Value of the object field that's used to resolve the procedure plan option.	Small, 62.0	62.0

Property Name	Type	Description	Filter Group and Version	Available Version
`fieldPath`	String	Path of the field that's used in a procedure in relation to the object that the field belongs to.	Small, 62.0	62.0
`isSuccess`	Boolean	Indicates whether the API request is successful (`true`) or not (`false`).	Small, 62.0	62.0
`literalValue`	String	User-defined value that's compared to the value of the sObject field value.	Small, 62.0	62.0
`operator`	String	Operator that's used by the procedure plan criterion.	Small, 62.0	62.0
`recordId`	String	ID of the procedure plan criterion record.	Small, 62.0	62.0

Procedure Plan Definition

Output representation of the details of a single procedure plan definition.

JSON example

This example shows a sample response for the Procedure Plan Definition By ID (GET) request.

```
        {
     "description": "Default Definition",
     "developerName": "Quote_Definition",
     "name": "Quote_Definition",
     "primaryObject": "Quote",
     "procedurePlanDefinitionVersions": [
     {
      "active": false,
      "contextDefinition": "11Oxx0000006PZ7EAM",
      "effectiveFrom": "2024-02-03T10:15:30.000Z",
      "effectiveTo": "2024-02-03T10:15:30.000Z",
      "readContextMapping": "MedicalHistoryMapping",
      "recordId": "1Cvxx0000004E1ACAU",
      "saveContextMapping": "MedicalHistoryMapping",
      "success": true
    }
  ],
  "recordId": "1FNxx0000004GkWGAU",
  "success": true
}
```

This example shows a sample response for the Procedure Plan Definition By ID (PATCH) request.

```
  {
  "procedurePlanDefinitionVersions":[],
  "recordId":"1FNDU00000000EX4AY",
  "success":true
  }
```

Property Name	Type	Description	Filter Group and Version	Available Version
description	String	Description for the procedure plan definition.	Small, 62.0	62.0
developerName	String	Developer name of the procedure plan definition.	Small, 62.0	62.0
error	Procedure Plan Generic Error[]	Details of the error encountered during the processing of the API request.	Small, 62.0	62.0
name	String	Name of the procedure plan definition.	Small, 62.0	62.0
primaryObject	String	Object that's associated with the procedure plan definition.	Small, 62.0	62.0
procedurePlanDefinitionVersions	Procedure Plan Definition Version[]	Details of the versions of a procedure plan definition.	Small, 62.0	62.0
recordId	String	ID of the procedure plan definition record.	Small, 62.0	62.0
success	Boolean	Indicates whether the API request is successful (true) or not (false).	Small, 62.0	62.0

Procedure Plan Definition Version

Output representation of the version details of a procedure plan definition.

JSON example

```
"procedurePlanDefinitionVersions": [
 {
   "active": false,
   "developerName": "sample_test",
   "effectiveFrom": "2024-07-09T00:00:00.000Z",
   "contextDefinition": "SalesTransactionContext__stdctx",
   "procedurePlanSections": [],
   "rank": 1,
   "readContextMapping": "ProductDiscoveryContextMapping",
   "recordId": "1CvZ60000008OIaKAM",
   "success": true
 }
]
```

Property Name	Type	Description	Filter Group and Version	Available Version
active	Boolean	Indicates whether the procedure plan definition version is active (true) or not (false).	Small, 62.0	62.0
contextDefinition	String	Context definition that's associated with the procedure plan definition version.	Small, 62.0	62.0

Property Name	Type	Description	Filter Group and Version	Available Version
developerName	String	Developer name of the procedure plan definition version.	Small, 62.0	62.0
effectiveFrom	String	Date and time from when the procedure plan definition version is effective.	Small, 62.0	62.0
effectiveTo	String	Date and time until when the procedure plan definition version is effective.	Small, 62.0	62.0
error	Procedure Plan Generic Error[]	Details of the error encountered during the processing of the API request.	Small, 62.0	62.0
inheritedFrom	String	Name of the template the procedure plan definition is extended from.	Small, 62.0	62.0
procedure PlanSections	Procedure Plan Section[]	List of sections of the procedure plan definition that you can organize in any order. Each section must include a procedure or a set of procedures to be executed for a specific criteria.	Small, 62.0	62.0
rank	Integer	Rank or the order of sequence to follow for the processing of the procedure plan definition version.	Small, 62.0	62.0
readContext Mapping	String	Mapping that's used to read data from the mapped object and populate the context definition.	Small, 62.0	62.0
recordId	String	ID of the procedure plan definition version record.	Small, 62.0	62.0
saveContext Mapping	String	Mapping that's used to save data from the context definition and populate the mapped object.	Small, 62.0	62.0
success	Boolean	Indicates whether the API request is successful (true) or not (false).	Small, 62.0	62.0

Procedure Plan Definitions

Output representation of the details of procedure plan definitions.

JSON example

```
{
            "isSuccess": true,
            "procedurePlanDefinitions": [
                {
                    "description": "test description",
                    "developerName": "sample_test",
                    "name": "sample_test",
```

```
                "primaryObject": "Account",
                "procedurePlanDefinitionVersions": [
                    {
                        "active": false,
                        "developerName": "sample_test",
                        "effectiveFrom": "2024-07-09T00:00:00.000Z",
                        "contextDefinition": "SalesTransactionContext__stdctx",
                        "procedurePlanSections": [],
                        "rank": 1,
                        "readContextMapping": "ProductDiscoveryContextMapping",
                        "recordId": "1CvZ60000008OIaKAM",
                        "success": true
                    }
                ],
                "recordId": "1FNZ60000004CAHOA2",
                "success": true
            },
            {
                "developerName": "PriceAdjustmentSchedule",
                "name": "PriceAdjustmentSchedule",
                "primaryObject": "PriceAdjustmentSchedule",
                "procedurePlanDefinitionVersions": [
                    {
                        "active": false,
                        "developerName": "PriceAdjustmentSchedule",
                        "effectiveFrom": "2024-07-10T00:00:00.000Z",
                        "contextDefinition": "SalesTransactionContext__stdctx",
                        "procedurePlanSections": [],
                        "rank": 1,
                        "recordId": "1CvZ6000000CaRbKAK",
                        "success": true
                    }
                ],
                "recordId": "1FNZ6000000CaSAOA0",
                "success": true
            }
            ]
        }
```


Property Name	Type	Description	Filter Group and Version	Available Version
`error`	Procedure Plan Generic Error[]	Details of the error encountered during the processing of the API request.	Small, 62.0	62.0
`isSuccess`	Boolean	Indicates whether the API request is successful (`true`) or not (`false`).	Small, 62.0	62.0
`procedurePlanDefinitions`	Procedure Plan Definition[]	Details of a single procedure plan definition.	Small, 62.0	62.0

Procedure Plan Generic

Output representation of the details of the created procedure plan definition record.

JSON example

This example shows a sample response of the details of a procedure plan definition record, created by using the Procedure Plan Definitions (POST) API.

```
{
 "isSuccess":true,
 "recordId":"1FNDU00000000EX4AY"
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`error`	Procedure Plan Generic Error[]	Details of the error encountered during the processing of the API request.	Small, 62.0	62.0
`isSuccess`	Boolean	Indicates whether the API request is successful (`true`) or not (`false`).	Small, 62.0	62.0
`recordId`	String	ID of the created procedure plan definition record.	Small, 62.0	62.0

Procedure Plan Generic Error

Output representation of the error details related to the procedure plan definitions.

Property Name	Type	Description	Filter Group and Version	Available Version
`errorCode`	String	Code indicating the type of error.	Small, 62.0	62.0
`message`	String	Message stating the reason for the error, if any.	Small, 62.0	62.0

Procedure Plan Option

Output representation of the details of a procedure plan option.

JSON example

```
                  "procedurePlanOptions": [
                    {
                      "expressionSetApiName": "Revenue_Mgmt_Default_Pricing_Procedure",
                      "expressionSetDefinition": "9QAZ60000004ECOOA2",
                   "expressionSetLabel": "Revenue Management Default Pricing Procedure",

                      "isSuccess": true,
                      "logic": "1 AND 2 AND 3",
                      "primaryObject": "Account",
                      "priority": 1,
```

```
                "procedurePlanCriterion": [
                  {
                    "conditionSequence": 1,
                    "dataType": "Text",
                    "fieldObject": "BillingCountry",
                    "fieldPath": "BillingCountry",
                    "isSuccess": true,
                    "literalValue": "test",
                    "operator": "Equals",
                    "recordId": "1FiZ60000004C9cKAE"
                   },
                   {
                    "conditionSequence": 2,
                    "dataType": "Text",
                    "fieldObject": "BillingPostalCode",
                    "fieldPath": "BillingPostalCode",
                    "isSuccess": true,
                    "literalValue": "pramit",
                    "operator": "Equals",
                    "recordId": "1FiZ60000004C9dKAE"
                    },
                    {
                      "conditionSequence": 3,
                      "dataType": "Date",
                      "fieldObject": "LastActivityDate",
                      "fieldPath": "LastActivityDate",
                      "isSuccess": true,
                      "literalValue": "2024-07-14",
                      "operator": "LessThan",
                      "recordId": "1FiZ60000004C9eKAE"
                     }
                   ],
                  "recordId": "1FYZ6000000000fOAA",
                  "saveContextMapping": "AssetToSalesTransactionMapping"
                   }
                 ]
```


Property Name	Type	Description	Filter Group and Version	Available Version
`error`	Procedure Plan Generic Error[]	Details of the error encountered during the processing of the API request.	Small, 62.0	62.0
`expressionSetApiName`	String	API name of the expression set.	Small, 62.0	62.0
`expressionSetDefinition`	String	Expression set definition that's associated with this procedure plan option record.	Small, 62.0	62.0
`expressionSetLabel`	String	Label of the expression set that's associated with this procedure plan option record.	Small, 62.0	62.0
`isSuccess`	Boolean	Indicates whether the API request is successful (`true`) or not (`false`).	Small, 62.0	62.0

Property Name	Type	Description	Filter Group and Version	Available Version
logic	String	Computation logic for the conditions applied to a procedure plan option.	Small, 62.0	62.0
primaryObject	String	Source object that's used to create a procedure with rule-based criteria.	Small, 62.0	62.0
priority	Integer	Priority for the specified criteria.	Small, 62.0	62.0
procedurePlanCriterion	Procedure Plan Criterion[]	Details of the rule-based criteria for the procedure.	Small, 62.0	62.0
readContextMapping	String	Mapping that's used to read from the mapped object and populate the context definition.	Small, 62.0	62.0
recordId	String	ID of the procedure plan option record.	Small, 62.0	62.0
saveContextMapping	String	Mapping that's used to save data from the context definition and populate the mapped object.	Small, 62.0	62.0

Procedure Plan Section

Output representation of the details of a procedure plan section.

JSON example

```
                "procedurePlanSections": [
            {
               "isInherited": false,
               "isSuccess": true,
               "procedurePlanOptions": [
                 {
                   "expressionSetApiName": "Revenue_Mgmt_Default_Pricing_Procedure",
                   "expressionSetDefinition": "9QAZ60000004ECOOA2",
                 "expressionSetLabel": "Revenue Management Default Pricing Procedure",

                   "isSuccess": true,
                   "logic": "1 AND 2 AND 3",
                   "primaryObject": "Account",
                   "priority": 1,
                   "procedurePlanCriterion": [
                     {
                       "conditionSequence": 1,
                       "dataType": "Text",
                       "fieldObject": "BillingCountry",
                       "fieldPath": "BillingCountry",
                       "isSuccess": true,
                       "literalValue": "test",
                       "operator": "Equals",
                       "recordId": "1FiZ60000004C9cKAE"
                      },
```

```
                    {
                     "conditionSequence": 2,
                     "dataType": "Text",
                     "fieldObject": "BillingPostalCode",
                     "fieldPath": "BillingPostalCode",
                     "isSuccess": true,
                     "literalValue": "pramit",
                     "operator": "Equals",
                     "recordId": "1FiZ60000004C9dKAE"
                     },
                     {
                       "conditionSequence": 3,
                       "dataType": "Date",
                       "fieldObject": "LastActivityDate",
                       "fieldPath": "LastActivityDate",
                       "isSuccess": true,
                       "literalValue": "2024-07-14",
                       "operator": "LessThan",
                       "recordId": "1FiZ60000004C9eKAE"
                      }
                    ],
                   "recordId": "1FYZ6000000000fOAA",
                   "saveContextMapping": "AssetToSalesTransactionMapping"
                    }
                    ],
                  "recordId": "1FRZ60000008OIAOA2",
                  "resolutionType": "RuleBased",
                  "sectionType": "PricingProcedure",
                  "sequence": 1,
                  "subSectionType": "PricingProcedure"
                }
            ]
```


Property Name	Type	Description	Filter Group and Version	Available Version
`error`	Procedure Plan Generic Error[]	Details of the error encountered during the processing of the API request.	Small, 62.0	62.0
`isInherited`	Boolean	Indicates whether the procedure plan section is inherited from a template (`true`) or not (`false`).	Small, 62.0	62.0
`isSuccess`	Boolean	Indicates whether the API request is successful (`true`) or not (`false`).	Small, 62.0	62.0
`procedurePlanOptions`	Procedure Plan Option[]	List of procedure plan options.	Small, 62.0	62.0
`recordId`	String	ID of the procedure plan option record.	Small, 62.0	62.0
`resolutionType`	String	Type of resolution that's used to filter the procedure.	Small, 62.0	62.0

Property Name	Type	Description	Filter Group and Version	Available Version
sectionType	String	Type of section. Valid values are: • PricingProcedure • ProductDiscoveryProcedure • ProductQualificationProcedure • PricingDiscoveryProcedure • DiscountSpreadServiceProcedure • RatingProcedure • Custom • RatingDiscoveryProcedure	Small, 62.0	62.0
sequence	Integer	Sequence that's followed for the processing of the procedures.	Small, 62.0	62.0
subSectionType	String	Subsection that's added to the procedure plan definition.	Small, 62.0	62.0

Procedure Plan Section Evaluation Runtime

Output representation of the results from the procedure plan evaluation.

JSON example

```
                "procedurePlanSections": [
                    {
                        "expressionSetApiName": "pricingProcedure_usageType_3",
                        "expressionSetDefinitionId": "9QAZ60000004Ef6OAE",
                        "expressionSetLabel": "pricingProcedure_usageType_3",
                        "sectionType": "PricingProcedure",
                        "sequence": 1,
                        "subSectionType": "Section1",
                        "usageType": "DefaultPricing"
                    },
                    {
                        "expressionSetApiName": "productQualification_usageType_3",
                        "expressionSetDefinitionId": "9QAZ60000004EfFOAU",
                        "expressionSetLabel": "productQualification_usageType_3",
                        "sectionType": "ProductQualificationProcedure",
                        "sequence": 3,
                        "subSectionType": "Section2",
                        "usageType": "ProductQualification"
                    },
                    {
                        "expressionSetApiName": "rating_usageType_2",
                        "expressionSetDefinitionId": "9QAZ60000004EfHOAU",
                        "expressionSetLabel": "rating_usageType_2",
                        "sectionType": "RatingProcedure",
                        "sequence": 2,
```

```
                "subSectionType": "Section3",
                "usageType": "DefaultRating"
            }
        ]
```

Property Name	Type	Description	Filter Group and Version	Available Version
expressionSetApiName	String	API name of the expression set.	Small, 62.0	62.0
expressionSetDefinitionId	String	ID of the expression set definition.	Small, 62.0	62.0
expressionSetLabel	String	Label of the expression set.	Small, 62.0	62.0
readContextMapping	String	Mapping that's used to read data from the mapped object and populate the context definition.	Small, 62.0	62.0
saveContextMapping	String	Mapping that's used to save data from the context definition and populate the mapped object.	Small, 62.0	62.0
sectionType	String	Name of the evaluated section. Valid values are: • PricingProcedure • ProductDiscoveryProcedure • ProductQualificationProcedure • PricingDiscoveryProcedure • DiscountSpreadServiceProcedure • RatingProcedure • Custom • RatingDiscoveryProcedure	Small, 62.0	62.0
sequence	Integer	Sequence that's followed for the processing of the procedures.	Small, 62.0	62.0
subSectionType	String	Name of the evaluated subsection.	Small, 62.0	62.0
usageType	String	Usage type of the procedure.	Small, 62.0	62.0

Procedure Plan Evaluation

Output representation of the evaluation details of a procedure plan definition.

JSON example

```
"procedurePlanEvaluations":[
{
  "errorMessage":"",
  "id":"a01DU000000BylcYAC",
  "isSuccess":true,
  "primaryObject":"SignallingCustomEvaluation__c",
  "result":{
  "contextDefinition":"11ODU00000008Sw2AI",
  "procedurePlanSections":[]
}
}
]
```

Property Name	Type	Description	Filter Group and Version	Available Version
`errorMessage`	String	Message indicating the error details, if any.	Small, 62.0	62.0
`id`	String	ID of the object used for evaluation.	Small, 62.0	62.0
`isSuccess`	Boolean	Indicates whether the API request is successful (`true`) or not (`false`).	Small, 62.0	62.0
`primaryObject`	String	Name of the object used for evaluation.	Small, 62.0	62.0
`result`	Procedure Plan Evaluation Result[]	Results from the procedure plan evaluation.	Small, 62.0	62.0

Procedure Plan Evaluation Response

Output representation of the evaluation details of a procedure plan definition.

JSON example

```
{
"isSuccess":true,
"procedurePlanEvaluations":[
{
  "errorMessage":"",
  "id":"a01DU000000BylcYAC",
  "isSuccess":true,
  "primaryObject":"SignallingCustomEvaluation__c",
  "result":{
  "contextDefinition":"11ODU00000008Sw2AI",
  "procedurePlanSections":[]
}
}
]
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
errorMessage	String	Message indicating the error details, if any.	Small, 62.0	62.0
isSuccess	Boolean	Indicates whether the API request is successful (true) or not (false).	Small, 62.0	62.0
procedurePlanDefinitionName	String	Name of the procedure plan definition.	Small, 62.0	62.0
procedurePlanEvaluations	Procedure Plan Evaluation[]	Evaluation details of the procedure plan.	Small, 62.0	62.0

Procedure Plan Evaluation Result

Output representation of the evaluation result of a procedure plan definition.

JSON example

```
    "result":{
    "contextDefinition":"11ODU00000008Sw2AI",
    "procedurePlanSections":[]
  }
```

Property Name	Type	Description	Filter Group and Version	Available Version
contextDefinition	String	Context definition that's associated with the procedure plan evaluation.	Small, 62.0	62.0
procedurePlanSections	Procedure Plan Section Evaluation Runtime[]	Results from the procedure plan evaluation.	Small, 62.0	62.0

Pricing Process Execution Details for Line Items

Output representation of the pricing process execution details for the line items along with the error details and response generation status.

JSON example

```
{
 "error": {},
 "isSuccess": true,
 "lineItemDetailsList": [
   {
     "lineItemId": "LineItem1",
     "status": "Success"
   },
   {
```

```
      "lineItemId": "LineItem2",
      "status": "Success"
    },
    {
      "lineItemId": "LineItem3",
      "status": "Failure"
    }
  ]
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`error`	Pricing Error Response	Error encountered during the processing of the API request.	Small, 63.0	63.0
`isSuccess`	Boolean	Indicates whether the response was generated successfully (`true`) or not (`false`).	Small, 63.0	63.0
`lineItemDetailsList`	Line Item Details Response []	List of the line items for which the pricing process is executed.	Small, 63.0	63.0

Line Item Details Response

Output representation of the pricing process execution details for the line items.

JSON example

```
{
 "lineItemDetailsList": [
   {
     "lineItemId": "LineItem1",
     "status": "Success"
   },
   {
     "lineItemId": "LineItem2",
     "status": "Success"
   },
   {
     "lineItemId": "LineItem3",
     "status": "Failure"
   }
  ]
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`lineItemId`	String	ID of the line item that the pricing process is executed for.	Small, 63.0	63.0

Property Name	Type	Description	Filter Group and Version	Available Version
status	String	Specifies whether the pricing process execution for the line item is successful or has failed. Valid values are: • Success • Failure	Small, 63.0	63.0

Pricing Process Execution Response

Output representation of the details of a pricing process execution.

JSON example

```
{
 "error": {},
 "isSuccess": true,
 "pricingProcessExecutionList": [
   {
     "executionId": "12345",
     "executionType": "Pricing_Line",
     "executionTypeId": "111_LineItem1",
     "message": "The Pricing API execution was successful.",
     "status": "Success"
   },
   {
     "executionId": "12345",
     "executionType": "Api_Execution",
     "executionTypeId": "333",
     "status": "Partial_Success"
   },
   {
     "executionId": "12345",
     "executionType": "Discovery",
     "executionTypeId": "222",
     "status": "Success"
   },
   {
     "executionId": "12345",
     "executionType": "Pricing",
     "executionTypeId": "111",
     "status": "Failure"
   },
   {
     "executionId": "12345",
     "executionType": "Discovery_Line",
     "executionTypeId": "222_LineItem1",
     "status": "Partial_Success"
   },
   {
```

```
      "executionId": "12345",
      "executionType": "Pricing_Line",
      "executionTypeId": "111_LineItem2",
      "status": "Failure"
    }
  ]
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`error`	Pricing Error Response	Error encountered during the processing of the API request.	Small, 63.0	63.0
`isSuccess`	Boolean	Indicates whether the response was generated successfully (`true`) or not (`false`).	Small, 63.0	63.0
`pricingProcessExecutionList`	Pricing Process Execution List []	List of the execution details of the pricing process.	Small, 63.0	63.0

Pricing Process Execution List

Output representation of the execution details for different types of the pricing processes.

JSON example

```
{
 "pricingProcessExecutionList": [
   {
     "executionId": "12345",
     "executionType": "Pricing_Line",
     "executionTypeId": "111_LineItem1",
     "message": "The Pricing API execution was successful.",
     "status": "Success"
   },
   {
     "executionId": "12345",
     "executionType": "Api_Execution",
     "executionTypeId": "333",
     "status": "Success"
   },
   {
     "executionId": "12345",
     "executionType": "Discovery",
     "executionTypeId": "222",
     "status": "Success"
   },
   {
     "executionId": "12345",
     "executionType": "Pricing",
     "executionTypeId": "111",
     "status": "Failure"
```

```
    },
    {
      "executionId": "12345",
      "executionType": "Discovery_Line",
      "executionTypeId": "222_LineItem1",
      "status": "Success"
    },
    {
      "executionId": "12345",
      "executionType": "Pricing_Line",
      "executionTypeId": "111_LineItem2",
      "status": "Failure"
    }
  ]
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`executionId`	String	Unique ID that's generated each time a pricing process is executed.	Small, 63.0	63.0
`executionType`	String	Type of the execution that's defined internally within the pricing API.	Small, 63.0	63.0
`executionTypeId`	String	Unique execution type ID that's generated internally for process executions, such as pricing or discovery procedures.	Small, 63.0	63.0
`message`	String	Message that's generated when a pricing process is executed.	Small, 63.0	63.0
`status`	String	Execution process status for a line item. Valid values are: • `Failure` • `Partial_Success`—Applies to `Pricing` and `Discovery` procedures when execution for some line items fails. • `Success`	Small, 63.0	63.0

Salesforce Pricing Standard Invocable Actions

Learn more about the standard invocable actions available with Salesforce Pricing.

Run Salesforce Headless Pricing Action

Invoke the Pricing Connect API by providing the pricing data and details of a context, pricing procedure, and price waterfall.

Run Salesforce Pricing Action

Invoke the Pricing Connect API by providing the context, pricing procedure, and price waterfall details.

SEE ALSO:

Actions Developer Guide: Overview

REST API Developer Guide: Invocable Actions Standard

Run Salesforce Headless Pricing Action

Invoke the Pricing Connect API by providing the pricing data and details of a context, pricing procedure, and price waterfall.

This action is available in API version 60.0 and later.

Special Access Rules

The Run Salesforce Headless Pricing action is available in Enterprise, Unlimited, and Developer Editions where Salesforce Pricing is enabled.

Supported REST HTTP Methods

URI

`/services/data/v63.0/actions/standard/runSalesforceHeadlessPricing`

Formats

JSON, XML

HTTP Methods

POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`contextDefinitionId`	**Type** string **Description** Required. ID of the context definition record that's used to build the pricing data.
`contextMappingId`	**Type** string **Description** Required.

Input	Details
	ID of the context-mapping record that identifies which Salesforce object and mappings to use to build the pricing data.
`discoveryProcedure`	**Type** string **Description** Name of the discovery procedure that's used to execute the discovery process.
`displayContext`	**Type** boolean **Description** Indicates whether to display the context structure for pricing (`true`) or not (`false`).
`effectiveDate`	**Type** string **Description** Date when the pricing rules, as specified in the pricing procedure, are applied. The `effectiveDate` parameter determines which pricing procedure to execute when multiple active versions of pricing procedures are available with different date ranges.
`isHighVolumeLineItems`	**Type** boolean **Description** Indicates whether the pricing API returns pricing details for more than 100 line items (`true`) or not (`false`).
`isSkipWaterfall`	**Type** boolean **Description** Indicates whether to generate the price waterfall data (`true`) or not (`false`).
`persistContext`	**Type** boolean **Description** Indicates whether to store the context (`true`) or not (`false`).
`pricingData`	**Type** string **Description** Required.

Input	Details
	JSON data that's used to build the pricing data. The JSON must be escaped.
`pricingProcedureId`	**Type** string **Description** Required. ID of the expression set definition record that's used to execute the pricing process.
`skipDiscovery`	**Type** boolean **Description** Indicates whether to skip executing the discovery procedure (`true`) or not (`false`).
`taggedData`	**Type** boolean **Description** Indicates whether the associated context node contains a key (`true`) or not (`false`).
`useSessionScopedContext`	**Type** boolean **Description** Indicates whether to store the context in a session (`true`) or a request (`false`).

Outputs

Output	Details
`contextDetails`	**Type** string **Description** Context structure for pricing that's generated when the `displayContext` parameter is set to `true`.
`executionId`	**Type** string **Description** ID of the executed pricing data.
`pricingProcessErrors`	**Type** string

Output	Details
	Description Errors that are generated due to context tags in a pricing process.
`pricingProcessStatus`	**Type** string **Description** Status of the pricing process, which is executed as an API call.
`pricingResult`	**Type** string **Description** Outcome of the executed pricing process that's based on the output tags defined in the associated context definition.

Example

POST

This sample request is for the Run Salesforce Headless Pricing action.

```
{
  "inputs": [
    {
      "contextMappingId": "11jSB000002Bn13YAC",
      "contextDefinitionId": "11OSB0000000WSv2AM",
      "pricingProcedureId": "9QLSB00000011T74AI",
      "discoveryProcedure": "ES1",
      "displayContext": true,
      "effectiveDate": "2023-11-16T12:20:00.000Z",
      "isHighVolumeLineItems": true,
      "skipDiscovery": true,
      "taggedData": false,
      "pricingData":
[illegible]

      "isSkipWaterfall": false,
      "persistContext": true,
      "useSessionScopedContext": false
    }
  ]
}
```

This sample response is for the Run Salesforce Headless Pricing action.

```
{
  "actionName": "runSalesforceHeadlessPricing",
  "errors": null,
  "isSuccess": true,
  "outputValues": {
```

```
  "contextDetails": {
    "SalesTransaction": [
      {
        "PriceBooks": "01sxx0000005q3VAAQ",
        "Subtotal": 587.2095,
        "PriceAdjustmentSchedule": "84Xxx0000004CMxEAM",
        "EffectiveDate": "2024-02-12T00:00:00.000Z",
        "SalesTransactionItem": [
          {
            "LineItemQuantity": 5,
            "ProductSellingModel": "0jPxx000000009hEAA",
            "DerivedPricingAttribute": false,
            "Product": "01txx0000006iLwAAI",
            "LineItem": "LineItem1",
            "NetUnitPrice": 117.4419,
            "SalesTrxnAdjustmentGroup": [
              {
                "AdjustmentType": "Percentage",
                "AdjustmentValue": 10
              }
            ]
          }
        ]
      }
    ]
  },
  "executionId": "1708488641379821",
  "pricingProcessStatus": "Completed",
  "pricingResult": {
    "StartDate": [
      {
        "value": 1700137200000,
        "dataPath": [
          "CTX_ece1667e-7a09-40ab-9718-23bdc179a0a5",
          "CTX_d3b9ffd5-2be7-4366-92d8-c2bcf03b69ed"
        ],
        "errors": [],
        "isSuccess": true
      }
    ],
    "NetUnitPrice": [
      {
        "value": 115,
        "dataPath": [
          "CTX_ece1667e-7a09-40ab-9718-23bdc179a0a5",
          "CTX_d3b9ffd5-2be7-4366-92d8-c2bcf03b69ed"
        ],
        "errors": [],
        "isSuccess": true
      }
    ],
    "ProductSellingModel": [
      {
        "value": "0jPDE000000042K2AQ",
```

```
        "dataPath": [
          "CTX_ece1667e-7a09-40ab-9718-23bdc179a0a5",
          "CTX_d3b9ffd5-2be7-4366-92d8-c2bcf03b69ed"
        ],
        "errors": [],
        "isSuccess": true
      }
    ],
    "Pricebook": [
      {
        "value": "01sDE0000000LoeYAE",
        "dataPath": [
          "CTX_ece1667e-7a09-40ab-9718-23bdc179a0a5"
        ],
        "errors": [],
        "isSuccess": true
      }
    ],
    "NetTotalPrice": [
      {
        "value": 1150,
        "dataPath": [
          "CTX_ece1667e-7a09-40ab-9718-23bdc179a0a5",
          "CTX_d3b9ffd5-2be7-4366-92d8-c2bcf03b69ed"
        ],
        "errors": [],
        "isSuccess": true
      }
    ],
    "Subtotal": [
      {
        "value": 1150,
        "dataPath": [
          "CTX_ece1667e-7a09-40ab-9718-23bdc179a0a5"
        ],
        "errors": [],
        "isSuccess": true
      }
    ],
    "Quantity": [
      {
        "value": 10,
        "dataPath": [
          "CTX_ece1667e-7a09-40ab-9718-23bdc179a0a5",
          "CTX_d3b9ffd5-2be7-4366-92d8-c2bcf03b69ed"
        ],
        "errors": [],
        "isSuccess": true
      }
    ],
    "Product": [
      {
        "value": "01tDE000000FU99YAG",
        "dataPath": [
```

```
            "CTX_ece1667e-7a09-40ab-9718-23bdc179a0a5",
            "CTX_d3b9ffd5-2be7-4366-92d8-c2bcf03b69ed"
          ],
          "errors": [],
          "isSuccess": true
        }
      ],
      "CurrencyIsoCode": [
        {
          "value": "USD",
          "dataPath": [
            "CTX_ece1667e-7a09-40ab-9718-23bdc179a0a5"
          ],
          "errors": [],
          "isSuccess": true
        }
      ]
    }
  },
  "version": 1
}
```


SEE ALSO:

Salesforce Help: Invoke Salesforce Headless Pricing in a Flow

Run Salesforce Pricing Action

Invoke the Pricing Connect API by providing the context, pricing procedure, and price waterfall details.

This action is available in API version 60.0 and later. You can use this action with Flows only. To use this action with an API tool such as Postman, see Run Salesforce Headless Pricing Action.

Special Access Rules

The Run Salesforce Pricing action is available in Developer, Enterprise, and Unlimited Editions where Salesforce Pricing is enabled.

Supported REST HTTP Methods

URI

`/services/data/v63.0/actions/standard/runSalesforcePricing`

Formats

JSON, XML

HTTP Methods

POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`contextInstanceId`	**Type** string **Description** Required. ID of the context data that's used to build the pricing procedure. Get the context instance ID by invoking the Context Service API. See Context Service (POST).
`discoveryProcedure`	**Type** string **Description** Name of the discovery procedure that's used to execute the discovery process of the pricing data.
`effectiveDate`	**Type** string **Description** Date when the pricing rules, as specified in the pricing procedure, are applied. The `effectiveDate` parameter determines which pricing procedure to execute when multiple active versions of pricing procedures are available with different date ranges.
`isDeveloperName`	**Type** boolean **Description** Indicates whether the input value in a procedure must use the API name of the pricing (`true`) or the field name (`false`).
`isSkipWaterfall`	**Type** boolean **Description** Indicates whether the price waterfall data must be generated (`true`) or not (`false`).
`pricingProcedureName`	**Type** string **Description** Required. Name of the pricing procedure record that's used to execute the pricing process.

Input	Details
`skipDiscovery`	**Type** boolean **Description** Indicates whether to skip executing the discovery procedure (`true`) or not (`false`).

Outputs

Output	Details
`executionId`	**Type** string **Description** ID of the executed pricing data.

Example

POST

This sample request is for the Run Salesforce Pricing action.

```
{
  "inputs": [
    {
      "contextInstanceId": "32f2c894-ba5e-41c0-91e4-2ab5826f579b",
      "pricingProcedureName": "PricingAction",
      "isSkipWaterfall": false,
      "skipDiscovery": false,
      "isDeveloperName": true,
      "effectiveDate": "2023-11-16T12:20:00.000Z",
      "discoveryProcedure": "ES1"
    }
  ]
}
```

This sample response is for the Run Salesforce Pricing action.

```
{
  "actionName": "runSalesforcePricing",
  "errors": null,
  "isSuccess": true,
  "outputValues": {
    "executionId": "2QTurzG2NRQ5bgrjvvqyh"
  },
```

```
  "version": 1
}
```

SEE ALSO:

Salesforce Help: Invoke Salesforce Pricing in a Flow

Salesforce Pricing Metadata API Types

Metadata API enables you to access some types and feature settings that you can customize in the user interface.

Flow for Salesforce Pricing

Represents the metadata associated with a flow. With Flow, you can create an application that navigates users through a series of screens to query and update records in the database. You can also execute logic and provide branching capability based on user input to build dynamic applications.

IndustriesPricingSettings

Represents the settings for Salesforce Pricing.

PricingActionParameters

Represents the pricing action that's associated with a context definition and pricing procedure.

PricingRecipe

Represents the data models or sets of objects of a particular cloud that the pricing data store consumes during design time and run time.

SEE ALSO:

Metadata API Developer Guide: Understanding Metadata API

Flow for Salesforce Pricing

Represents the metadata associated with a flow. With Flow, you can create an application that navigates users through a series of screens to query and update records in the database. You can also execute logic and provide branching capability based on user input to build dynamic applications.

FlowActionCall

Salesforce Pricing exposes additional actionType values for the FlowActionCall Metadata type.

Field Name	Field Type	Description
`actionType`	InvocableActionType (enumeration of type string)	Required. The action type. Additional valid values only for Salesforce Pricing include: • `runSalesforcePricing`—Invoke the Pricing Connect API by providing the context, pricing procedure, and price waterfall details.

Field Name	Field Type	Description
		• `runSalesforceHeadlessPricing`—Invoke the Pricing Connect API by providing the pricing data and details of a context, pricing procedure, and price waterfall.

IndustriesPricingSettings

Represents the settings for Salesforce Pricing.

Parent Type and Manifest Access

This type extends the Metadata metadata type and inherits its `fullName` field.

In the package manifest, all the settings metadata types for the org are accessed using the "Settings" name. See Settings for more details.

File Suffix and Directory Location

`IndustriesPricingSettings` values are stored in the `IndustriesPricing.settings` file in the `settings` folder. The `.settings` files are different from other named components, because there's only one settings file for each settings component.

Version

IndustriesPricingSettings components are available in API version 60.0 and later.

Special Access Rules

This metadata type is available with Salesforce Pricing.

Fields

Field Name	Description
`enableHighAvailability`	**Field Type** boolean **Description** Reserved for internal use.
`enableLowestPriceCompliance`	**Field Type** boolean **Description** Indicates whether to provide end buyer visibility on lowest price for 30 days (`true`) or not (`false`). The default value is `false`. Available in API version 62.0 and later.
`enablePricingWaterfall`	**Field Type** boolean

Field Name	Description
	Description Indicates whether to enable Price Waterfall (`true`) or not (`false`). The default value is `false`. Price Waterfall provides insights that include price breakups and reasons for every step of the pricing process.
`enablePricingWaterfallPersistence`	**Field Type** boolean **Description** Indicates whether to enable Price Waterfall Persistence (`true`) or not (`false`). The default value is `false`. Price Waterfall Persistence stores the process logs that provide insights into the internal pricing processes.
`enableSalesforcePricing`	**Field Type** boolean **Description** Indicates whether to enable Salesforce Pricing (`true`) or not (`false`). The default value is `false`.

Declarative Metadata Sample Definition

This example shows a sample IndustriesPricingSettings component.

```
<IndustriesPricingSettings xmlns="http://soap.sforce.com/2006/04/metadata">
     <enableHighAvailability>true</enableHighAvailability>
     <enableLowestPriceCompliance>true</enableLowestPriceCompliance>
     <enablePricingWaterfall>true</enablePricingWaterfall>
     <enablePricingWaterfallPersistence>true</enablePricingWaterfallPersistence>
     <enableSalesforcePricing>true</enableSalesforcePricing>
</IndustriesPricingSettings>
```

This example shows a sample `package.xml` that references the previous definition.

```
<?xml version="1.0" encoding="UTF-8"?>
<Package xmlns="http://soap.sforce.com/2006/04/metadata">
    <types>
        <members>IndustriesPricing</members>
        <name>Settings</name>
    </types>
    <version>63.0</version>
</Package>
```

Wildcard Support in the Manifest File

The wildcard character `*` (asterisk) in the `package.xml` manifest file doesn't apply to metadata types for feature settings. The wildcard applies only when retrieving all settings, not for an individual setting. For details, see Settings. For information about using the manifest file, see Deploying and Retrieving Metadata with the Zip File.

PricingActionParameters

Represents the pricing action that's associated with a context definition and pricing procedure.

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.

Parent Type

This type extends the Metadata metadata type and inherits its `fullName` field.

File Suffix and Directory Location

PricingActionParameters components have the suffix `.pricingActionParameters` and are stored in the `pricingActionParameters` folder.

Version

PricingActionParameters components are available in API version 60.0 and later.

Special Access Rules

This metadata type is available with Salesforce Pricing.

Fields

Field Name	Description
`contextDefinition`	**Field Type** string **Description** Required. Context definition record that's associated with the pricing action.
`contextMapping`	**Field Type** string **Description** Required. Context mapping record that's associated with the pricing action.
`developerName`	**Field Type** string **Description** Required. Unique name of the pricing action parameter record.

Field Name	Description
	The name must begin with a letter and use only alphanumeric characters and underscores. The name must not include spaces, end with an underscore, or have two consecutive underscores.
`effectiveFrom`	**Field Type** dateTime **Description** Required. Date and time from when the pricing action becomes effective.
`effectiveTo`	**Field Type** dateTime **Description** Date and time till when the pricing action is in effect.
`masterLabel`	**Field Type** string **Description** Required. Master label of the pricing action parameter.
`objectName`	**Field Type** string **Description** Name of the object that's associated with the pricing action. Valid values are: • `Case` • `Contract` • `Opportunity` • `Order` • `Quote` • `SalesAgreement` • `WorkOrder`
`pricingProcedure`	**Field Type** string **Description** Pricing procedure record that's associated with this pricing action.

Declarative Metadata Sample Definition

The following is an example of a PricingActionParameters component.

```
<PricingActionParameters xmlns="http://soap.sforce.com/2006/04/metadata">
    <developerName>CMEDefaultActionParameters</developerName>
    <objectName>ORDER</objectName>
    <pricingProcedure>PP</pricingProcedure>
    <effectiveFrom>2024-04-08T07:32:00.000Z</effectiveFrom>
    <effectiveTo>2024-04-11T07:32:00.000Z</effectiveTo>
    <contextDefinition>SalesTransactionContext__stdctx</contextDefinition>
    <contextMapping>SalesAgreementEntitiesMapping</contextMapping>
    <masterLabel>PAP_test</masterLabel>
</PricingActionParameters>
```

The following is an example `package.xml` that references the previous definition.

```
<?xml version="1.0" encoding="UTF-8"?>
<Package xmlns="http://soap.sforce.com/2006/04/metadata">
    <types>
        <members>*</members>
        <name>PricingActionParameters</name>
    </types>
    <version>63.0</version>
</Package>
```

Wildcard Support in the Manifest File

This metadata type supports the wildcard character `*` (asterisk) in the `package.xml` manifest file. For information about using the manifest file, see Deploying and Retrieving Metadata with the Zip File.

PricingRecipe

Represents the data models or sets of objects of a particular cloud that the pricing data store consumes during design time and run time.

Parent Type

This type extends the Metadata metadata type and inherits its `fullName` field.

File Suffix and Directory Location

PricingRecipe components have the suffix `.pricingRecipe` and are stored in the `pricingRecipe` folder.

Version

PricingRecipe components are available in API version 60.0 and later.

Special Access Rules

This metadata type is available with Salesforce Pricing.

Fields

Field Name	Description
`defaultPricingProcedure`	**Field Type** ExpressionSetDefinition **Description** Expression set definition that's associated with this pricing recipe setting.
`defaultPricingProcedureDeveloperName`	**Field Type** string **Description** For internal use only.
`defaultPricingProcedureId`	**Field Type** string **Description** ID of the pricing procedure of the pricing recipe.
`developerName`	**Field Type** string **Description** Required. API name of the pricing recipe.
`isActive`	**Field Type** boolean **Description** Indicates whether the pricing recipe is active (`true`) or not (`false`). The default value is `false`
`isInternal`	**Field Type** boolean **Description** Indicates whether the price recipe record is created internally by the Salesforce platform (`true`) or not (`false`). The default value is `false`
`masterLabel`	**Field Type** string **Description** Required. Name for pricing recipe that's defined when the pricing recipe is created.

Field Name	Description
pricingRecipeTableMapping	**Field Type** PricingRecipeTableMapping[] **Description** Mapping of the pricing components of a lookup table with the chosen pricing recipe.

PricingRecipeTableMapping

Represents the mapping of the lookup table with the chosen pricing recipe.

Field Name	Description
isInternal	**Field Type** boolean **Description** Indicates whether the price recipe field mapping record is created internally by the Salesforce platform (`true`) or not (`false`). The default value is `false`.
lookupTable	**Field Type** DecisionTable DecisionMatrixDefinition **Description** Lookup table that's associated with either a decision matrix or decision table.
lookupTableDeveloperName	**Field Type** string **Description** For internal use only.
pricingComponentType	**Field Type** string **Description** Pricing component field data that the decision table is built on. Valid values are: • `AttributeDiscount` • `BundleDiscount` • `DerivedPricing` • `ListPrice` • `PriceAdjustmentMatrix` • `PromotionsDiscount`

Field Name	Description
	• `VolumeDiscount` • `VolumeTierDiscount` • `DiscountDistributionService`. This value is available in API version 60.0 and later. • `MinimumPrice`. Available in API version 62.0 and later.
`pricingProcedureOutputMapList`	**Field Type** PricingProcedureOutputMap[] **Description** List of the mappings of the outputs of the pricing procedures to the associated lookup tables. Available in API version 60.0 and later.
`pricingRecipe`	**Field Type** string **Description** Required. Pricing data store that's associated with this pricing recipe field mapping.

PricingProcedureOutputMap

Represents the mapping of the outputs of the pricing procedures to the associated lookup tables. Each record specifies the output mapping of the associated lookup table based on the pricing component type specified in the PricingRecipeTableMapping object.

Field Name	Description
`fieldName`	**Field Type** string **Description** For internal use only.
`isPricingRecipeActive`	**Field Type** boolean **Description** Indicates whether the associated pricing recipe is active (`true`) or not (`false`). The default value is `false`.
`outputFieldName`	**Field Type** string **Description** Field name that contains the output type that's generated from the pricing element.

Field Name	Description
outputFieldNameString	**Field Type** string **Description** Derived field that references a specific column in a decision table or decision matrix.
outputType	**Field Type** string **Description** Output type that's generated from a pricing element. Valid values are: • AdjustmentType • AdjustmentValue • CustomOutput • HashOutput • UnitPrice

Declarative Metadata Sample Definition

The following is an example of a PricingRecipe component.

```
<PricingRecipe xmlns="http://soap.sforce.com/2006/04/metadata">
    <defaultPricingProcedureId> </defaultPricingProcedureId>
    <developerName>CMEDefaultRecipe</developerName>
    <isActive>false</isActive>
    <isInternal>false</isInternal>
    <masterLabel>CMEDefaultRecipe</masterLabel>
    <pricingRecipeTableMapping>
        <isInternal>false</isInternal>

<lookupTableDeveloperName>Bundle_Based_Adjustment_Decision_Table</lookupTableDeveloperName>

        <pricingComponentType>CUSTOMDISCOUNT</pricingComponentType>
        <fileBasedDecisionTableName>Bundle Based Adjustment
Entries</fileBasedDecisionTableName>
        <pricingProcedureOutputMapList>
            <fieldName>AdjustmentValue</fieldName>
            <isPricingRecipeActive>false</isPricingRecipeActive>
            <outputFieldName>01Pxx000000000f</outputFieldName>
            <outputFieldNameString>false</outputFieldNameString>
            <outputType>AdjustmentValue</outputType>
        </pricingProcedureOutputMapList>
        <pricingProcedureOutputMapList>
            <fieldName>AdjustmentType</fieldName>
            <isPricingRecipeActive>false</isPricingRecipeActive>
            <outputFieldName>01Pxx000000000m</outputFieldName>
            <outputFieldNameString>false</outputFieldNameString>
```

```
            <outputType>AdjustmentType</outputType>
        </pricingProcedureOutputMapList>
        <pricingRecipe>CMEDefaultRecipe</pricingRecipe>
    </pricingRecipeTableMapping>
</PricingRecipe>
```

The following is an example `package.xml` that references the previous definition.

```
<?xml version="1.0" encoding="UTF-8"?>
<Package xmlns="http://soap.sforce.com/2006/04/metadata">
    <types>
        <members>*</members>
        <name>PricingRecipe</name>
    </types>
    <version>63.0</version>
</Package>
```

Wildcard Support in the Manifest File

This metadata type supports the wildcard character `*` (asterisk) in the `package.xml` manifest file. For information about using the manifest file, see Deploying and Retrieving Metadata with the Zip File.

Salesforce Pricing Tooling API Objects

Tooling API exposes metadata used in developer tooling that you can access through REST or SOAP. Tooling API's SOQL capabilities for many metadata types allow you to retrieve smaller pieces of metadata.

PricingActionParameters

Represents a pricing action associated to a context definition and a pricing procedure. This object is available in API version 60.0 and later.

PricingProcedureOutputMap

Represents the mapping of the outputs of the pricing procedures to the associated lookup tables. Each record specifies the output mapping of the associated lookup table based on the pricing component type specified in the Pricing Recipe Table Mapping object. This object is available in API version 60.0 and later.

PricingRecipe

Represents one out of various data models or sets of entities of a particular cloud that'll be consumed by the pricing data store during design and run time. This object is available in API version 60.0 and later.

PricingRecipeTableMapping

Represents the mapping of pricing components of a lookup table with the chosen pricing recipe. This object is available in API version 60.0 and later.

ProcedureOutputResolution

Represents the pricing resolution for an pricing element determined using strategy name and formula. This object is available in API version 63.0 and later.

ProcedurePlanCriterion

Represents a criterion within a procedure plan option record. This object is available in API version 62.0 and later.

ProcedurePlanDefinition

Represents the setup of a unified procedure from a list of multiple procedures that can be sequenced in any order based on business needs. Each procedure plan definition contains sections and subsections where procedures can be configured by using a lookup table or rule-based criteria. This object is available in API version 62.0 and later.

ProcedurePlanDefinitionVersion

Represents the versions for a procedure plan definition. Multiple versions under a procedure plan definition must be active at a time, which can be resolved at run time using the rank field. This object is available in API version 62.0 and later.

ProcedurePlanOption

Represents the selection criteria of how a procedure can be configured for a selected procedure plan section record. This object is available in API version 62.0 and later.

ProcedurePlanSection

Represents various procedure setup sections for a procedure plan definition. Each section enables the setup of a procedure of a type that can be further determined by using a rule-based criteria or it can be set based on a selected lookup table. This object is available in API version 62.0 and later.

ProcedurePlanVariable

Represents the setup for any adhoc user-defined variable that can be linked to a procedure plan definition record. This object is available in API version 62.0 and later.

SEE ALSO:

Tooling API Developer Guide: Introducing Tooling API

PricingActionParameters

Represents a pricing action associated to a context definition and a pricing procedure. This object is available in API version 60.0 and later.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field	Details
`ContextDefinition`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** The context definition record associated with the pricing action.
`ContextMapping`	**Type** string

Field	Details
	Properties Create, Filter, Group, Sort, Update **Description** The context mapping record that's associated with the pricing action.
`DeveloperName`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** The unique name of the pricing action parameter record. This name must begin with a letter and use only alphanumeric characters and underscores. It can't include spaces, end with an underscore, or have two consecutive underscores.
`EffectiveFrom`	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** The date and time when the pricing action comes into effect.
`EffectiveTo`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date and time till when the pricing action is in effect.
`Language`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The language of the pricing action parameter. Possible values are: • `da`—Danish • `de`—German • `en_US`—English • `es`—Spanish • `es_MX`—Spanish (Mexico)

Field	Details
	• `fi`—Finnish • `fr`—French • `it`—Italian • `ja`—Japanese • `ko`—Korean • `nl_NL`—Dutch • `no`—Norwegian • `pt_BR`—Portuguese (Brazil) • `ru`—Russian • `sv`—Swedish • `th`—Thai • `zh_CN`—Chinese (Simplified) • `zh_TW`—Chinese (Traditional)
`MasterLabel`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** The master label of the pricing action parameter.
`NamespacePrefix`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The namespace prefix that is associated with this object. Each Developer Edition org that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the namespacePrefix__componentName notation. The namespace prefix can have one of the following values. • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There's an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are Developer Edition organizations, NamespacePrefix is only set for objects that are part of an installed managed package. There's no namespace prefix for all other objects.

Field	Details
`ObjectName`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Specifies the object associated to the pricing action. Possible values are: • `Case` • `Contract` • `Opportunity` • `Order` • `Quote` • `SalesAgreement` • `WorkOrder`
`PricingProcedure`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The pricing procedure record associated with this pricing action.

PricingProcedureOutputMap

Represents the mapping of the outputs of the pricing procedures to the associated lookup tables. Each record specifies the output mapping of the associated lookup table based on the pricing component type specified in the Pricing Recipe Table Mapping object. This object is available in API version 60.0 and later.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field	Details
`IsPricingRecipeActive`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort

Field	Details
	Description Indicates if the pricing recipe is active (true) or not (false). The default value is `false`.
`LookupField`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** Definition of the fields that are used for this lookup.
`OutputFieldNameId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The field name containing the output type generated from the pricing element. This field is a polymorphic relationship field. **Relationship Name** OutputFieldName **Relationship Type** Lookup **Refers To** CalculationMatrixColumn, DecisionTableParameter
`OutputFieldNameString`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** This is a derived field that references a specific column in a decision table or decision matrix.
`OutputType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The output type generated from a pricing element. Possible values are: • `AdjustmentType`—Adjustment Type

Field	Details
	• `AdjustmentValue`—Adjustment Value • `CustomOutput`—Custom Output • `HashOutput`—Hash Output • `UnitPrice`—Unit Price
`PricingComponentType`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** The pricing component field data on which the decision table is built. Possible values are: • `AttributeDiscount`—Attribute Based Discount • `BundleDiscount`—Bundle Based Discount • `DerivedPricing` • `DiscountDistributionService`—Discount Distribution Service • `ListPrice`—List Price • `PriceAdjustmentMatrix` • `PromotionsDiscount` • `VolumeDiscount`—Volume Based Discount • `VolumeTierDiscount`—Tier Discount
`PricingRecipeTableMappingId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The mapping of pricing components of a lookup table with the chosen pricing recipe. This field is a relationship field. **Relationship Name** PricingRecipeTableMapping **Relationship Type** Lookup **Refers To** PricingRecipeTableMapping

PricingRecipe

Represents one out of various data models or sets of entities of a particular cloud that'll be consumed by the pricing data store during design and run time. This object is available in API version 60.0 and later.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

Fields

Field	Details
`DefaultPricingProcedureId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The expression set definition or Salesforce flow definition associated with this pricing recipe settings. This field is a relationship field. **Relationship Name** DefaultPricingProcedure **Relationship Type** Lookup **Refers To** ExpressionSetDefinition
`DeveloperName`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** The developer name of the pricing recipe.
`IsActive`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the pricing recipe is active (true) or not (false).

Field	Details
	The default value is `false`.
`IsInternal`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Indicates if the price recipe record is created internally by the Salesforce platform (true) or not (false). The default value is `false`.
`Language`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The languages in which pricing recipe is supported. Possible values are: • `da`—Danish • `de`—German • `en_US`—English • `es`—Spanish • `es_MX`—Spanish (Mexico) • `fi`—Finnish • `fr`—French • `it`—Italian • `ja`—Japanese • `ko`—Korean • `nl_NL`—Dutch • `no`—Norwegian • `pt_BR`—Portuguese (Brazil) • `ru`—Russian • `sv`—Swedish • `th`—Thai • `zh_CN`—Chinese (Simplified) • `zh_TW`—Chinese (Traditional)
`MasterLabel`	**Type** string

Field	Details
	Properties Create, Filter, Group, Sort, Update **Description** A user-friendly name for pricing recipe, which is defined when the pricing recipe is created.
`NamespacePrefix`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The namespace prefix that is associated with this object. Each Developer Edition org that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the namespacePrefix__componentName notation. The namespace prefix can have one of the following values. • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There's an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are Developer Edition organizations, NamespacePrefix is only set for objects that are part of an installed managed package. There's no namespace prefix for all other objects.

PricingRecipeTableMapping

Represents the mapping of pricing components of a lookup table with the chosen pricing recipe. This object is available in API version 60.0 and later.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field	Details
`FileBasedDecisionTableName`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description Name of the file-based decision table.
`IsInternal`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Indicates if the price recipe field mapping record is created internally by the Salesforce platform (true) or not (false). The default value is `false`.
`LookupTableId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The lookup table associated with the mapped fields. This field is a polymorphic relationship field. **Relationship Name** LookupTable **Relationship Type** Lookup **Refers To** DecisionMatrixDefinition, DecisionTable
`PricingComponentType`	**Type** Pricing Element Type enumerated list **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The pricing component field data on which the decision table is built. Possible values are: • `AttributeDiscount`—Attribute Based Discount • `BundleDiscount`—Bundle Based Discount • `DerivedPricing` • `ListPrice`—List Price • `PriceAdjustmentMatrix` • `PromotionsDiscount` • `VolumeDiscount`—Volume Based Discount

Field	Details
	• `VolumeTierDiscount`—Tier Discount • `DiscountDistributionService`. This value is available in API version 60.0 and later. • `MinimumPrice`. This value is available in API version 62.0 and later.
`PricingRecipeId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The pricing data store associated with this pricing recipe field mappings. This field is a relationship field. **Relationship Name** PricingRecipe **Relationship Type** Lookup **Refers To** PricingRecipe

ProcedureOutputResolution

Represents the pricing resolution for an pricing element determined using strategy name and formula. This object is available in API version 63.0 and later.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Fields

Field	Details
`DeveloperName`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** The name of the procedure output resolution.
`Formula`	**Type** string

Field	Details
	Properties Create, Filter, Group, Sort, Update **Description** The formula function used to determine the minimum or maximum price of a product. The supported operations are MIN and MAX.
`IsActive`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the procedure output resolution is active (true) or not (false). Only active procedure output resolutions can be applied to a procedure. The default value is `false`.
`IsInternal`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort **Description** Indicates if the procedure output resolution record is created internally by the Salesforce platform (true) or not (false). The default value is `false`.
`Language`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The languages in which pricing recipe is supported. Possible values are: • `da`—Danish • `de`—German • `en_US`—English • `es`—Spanish • `es_MX`—Spanish (Mexico) • `fi`—Finnish • `fr`—French • `it`—Italian • `ja`—Japanese

Field	Details
	• `ko`—Korean • `nl_NL`—Dutch • `no`—Norwegian • `pt_BR`—Portuguese (Brazil) • `ru`—Russian • `sv`—Swedish • `th`—Thai • `zh_CN`—Chinese (Simplified) • `zh_TW`—Chinese (Traditional)
`MasterLabel`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** A user-friendly name for procedure output resolution, which is defined when the procedure output resolution record is created.
`NamespacePrefix`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The namespace prefix that is associated with this object. Each Developer Edition org that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the namespacePrefix__componentName notation. The namespace prefix can have one of the following values. • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There's an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are Developer Edition organizations, NamespacePrefix is only set for objects that are part of an installed managed package. There's no namespace prefix for all other objects.
`PricingElement`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update

Field	Details
	Description Specifies the pricing element on which the procedure output resolution is defined. Possible values are: • `ListPrice`—List Price • `MinimumPrice`—Price Tracking

ProcedurePlanCriterion

Represents a criterion within a procedure plan option record. This object is available in API version 62.0 and later.

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.

Supported SOAP API Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Supported REST API Methods

`DELETE, GET, HEAD, PATCH, POST, Query`

Fields

Field	Details
`ActualValue`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The user-defined value that's compared to the value of the sObject field value.
`DataType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The data type of the field from the selected object.
`FieldPath`	**Type** string

Field	Details
	Properties Create, Filter, Group, Sort, Update **Description** The path of the field used in a procedure in relation to the object that the field belongs to.
`ObjectField`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** The Salesforce object field value used to resolve the procedure plan option.
`Operator`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The operator used by the procedure plan criterion. Valid values are: • `Equals` • `GreaterThan` • `GreaterThanOrEquals` • `In` • `IsNotNull` • `IsNull` • `LessThan` • `LessThanOrEquals` • `NotEquals` • `NotIn`
`ProcedurePlanOptionId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The procedure plan option associated with the procedure plan criterion record. This field is a relationship field. **Relationship Name** ProcedurePlanOption

Field	Details
	Relationship Type Master-detail **Refers To** ProcedurePlanOption (the master object)
`Sequence`	**Type** int **Properties** Create, Filter, Group, Sort **Description** The sequence in which the conditions defined in the procedure plan criteria are processed.

ProcedurePlanDefinition

Represents the setup of a unified procedure from a list of multiple procedures that can be sequenced in any order based on business needs. Each procedure plan definition contains sections and subsections where procedures can be configured by using a lookup table or rule-based criteria. This object is available in API version 62.0 and later.

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.

Supported SOAP API Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Supported REST API Methods

`DELETE, GET, HEAD, PATCH, POST, Query`

Fields

Field	Details
`Description`	**Type** textarea **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The description of the procedure plan definition.
`DeveloperName`	**Type** string

Field	Details
	Properties Create, Filter, Group, Sort, Update **Description** The unique name of the procedure plan definition record. This name must begin with a letter and use only alphanumeric characters and underscores. It can't include spaces, end with an underscore, or have two consecutive underscores.
`Language`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The language of the procedure plan definition. Valid values are: • `da`—Danish • `de`—German • `en_US`—English • `es`—Spanish • `es_MX`—Spanish (Mexico) • `fi`—Finnish • `fr`—French • `it`—Italian • `ja`—Japanese • `ko`—Korean • `nl_NL`—Dutch • `no`—Norwegian • `pt_BR`—Portuguese (Brazil) • `ru`—Russian • `sv`—Swedish • `th`—Thai • `zh_CN`—Chinese (Simplified) • `zh_TW`—Chinese (Traditional)
`MasterLabel`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** The master label of the procedure plan definition.

Field	Details
`NamespacePrefix`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The namespace prefix that is associated with this object. Each Developer Edition org that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the namespacePrefix__componentName notation. The namespace prefix can have one of the following values. • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There's an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are Developer Edition organizations, NamespacePrefix is only set for objects that are part of an installed managed package. There's no namespace prefix for all other objects.
`PrimaryObject`	**Type** string **Properties** Create, Filter, Group, idLookup, Nillable, Sort **Description** The object associated to the procedure plan definition. The fields in the object are used as variables in the procedure plan criterion.
ProcessType	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Identify the business processes that need a procedure plan for each SObject and definition. Possible values are: • `Default` • `ProductDiscovery`—Product Discovery The default value is `Default`.

ProcedurePlanDefinitionVersion

Represents the versions for a procedure plan definition. Multiple versions under a procedure plan definition must be active at a time, which can be resolved at run time using the rank field. This object is available in API version 62.0 and later.

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.

Supported SOAP API Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Supported REST API Methods

`DELETE, GET, HEAD, PATCH, POST, Query`

Fields

Field	Details
`ContextDefinition`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The context definition associated with the procedure plan definition record. Valid values are: • `11ODU00000007Zx2AI` • `11ODU000000084F2AQ` • `CollectionPlanEvent__stdctx` • `CommerceCartContextDefinition__stdctx` • `SalesTransactionContext__stdctx` • `TestContextService__stdctx` • `TestDynamicAttribute__stdctx` • `TestExtendedDefinition__stdctx`
`DefaultReadContextMapping`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The default read context mapping used to read from the mapped object and populate the context definition.

Field	Details
DefaultSaveContextMapping	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The save context mapping used to save from the context definition and populate the mapped object.
DeveloperName	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** The unique name of the procedure plan definition version record. This name must begin with a letter and use only alphanumeric characters and underscores. It can't include spaces, end with an underscore, or have two consecutive underscores.
EffectiveFrom	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** The date and time when the procedure plan definition comes into effect.
EffectiveTo	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date and time when the procedure plan definition is no longer in effect.
InheritedFrom	**Type** string **Properties** Create, Filter, Group, idLookup, Nillable, Sort **Description** The template from which this procedure plan definition is created.
IsActive	**Type** boolean

Field	Details
	Properties Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates if this procedure plan definition version is active (`true`) or not (`false`). The default value is `false`.
`ProcedurePlanDefinitionId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The procedure plan definition associated with this procedure plan definition version record. This field is a relationship field. **Relationship Name** ProcedurePlanDefinition **Relationship Type** Master-detail **Refers To** ProcedurePlanDefinition (the master object)
`Rank`	**Type** int **Properties** Create, Filter, Group, Sort, Update **Description** The current rank of the procedure plan definition version that's used to determine which procedure plan definition version is executed .

ProcedurePlanOption

Represents the selection criteria of how a procedure can be configured for a selected procedure plan section record. This object is available in API version 62.0 and later.

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.

Supported SOAP API Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Supported REST API Methods

`DELETE, GET, HEAD, PATCH, POST, Query`

Fields

Field	Details
`CriteriaLogic`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The computation logic for the various conditions applied to an option.
`CtxDefinitionOutputFieldId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The context definition field that's associated with the decision table. This field is a relationship field. **Relationship Name** CtxDefinitionOutputField **Refers To** DecisionTableParameter
`CtxMappingOutputFieldId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The context mapping field that's associated with the decision table. This field is a relationship field. **Relationship Name** CtxMappingOutputField **Refers To** DecisionTableParameter
`DecisionTableId`	**Type** reference **Properties** Filter, Group, Nillable, Sort

Field	Details
	Description The decision table associated with the pricing procedure. This field is a relationship field. **Relationship Name** DecisionTable **Refers To** DecisionTable
`ExpressionSetApiName`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The API name of the expression set.
`ExpressionSetDefinitionId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The expression set definition associated with the procedure plan option record. This field is a relationship field. **Relationship Name** ExpressionSetDefinition **Refers To** ExpressionSetDefinition
`ExpressionSetLabel`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The label of the expression set definition.
`ExpressionSetOutputFieldId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The expression set output field that's associated with the decision table. This field is a relationship field.

Field	Details
	Relationship Name ExpressionSetOutputField **Refers To** DecisionTableParameter
`PrimaryObject`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The procedure plan definition associated with the procedure plan option record.
`Priority`	**Type** int **Properties** Create, Filter, Group, Sort **Description** The order in which the options are executed.
`ProcedurePlanSectionId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The procedure plan section associated with the procedure plan option record. This field is a relationship field. **Relationship Name** ProcedurePlanSection **Relationship Type** Master-detail **Refers To** ProcedurePlanSection (the master object)
`ReadContextMapping`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The read context mapping used to read from the mapped object and populate the context definition.

Field	Details
SaveContextMapping	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The save context mapping used to save from the context definition and populate the mapped object.

ProcedurePlanSection

Represents various procedure setup sections for a procedure plan definition. Each section enables the setup of a procedure of a type that can be further determined by using a rule-based criteria or it can be set based on a selected lookup table. This object is available in API version 62.0 and later.

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.

Supported SOAP API Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Supported REST API Methods

`DELETE, GET, HEAD, PATCH, POST, Query`

Fields

Field	Details
Description	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The description of the procedure plan section.
IsInherited	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort **Description** Indicates if the section is inherited from a template (`true`) ot not (`false`).

Field	Details
	The default value is `false`.
`ProcedurePlanVersionId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The procedure plan version associated with this procedure plan section record. This field is a relationship field. **Relationship Name** ProcedurePlanVersion **Relationship Type** Master-detail **Refers To** ProcedurePlanDefinitionVersion (the master object)
`ResolutionType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The type of resolution used to filter the procedure. Valid values are: • `Default` • `RuleBased`
`SectionType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort **Description** The type of procedure section. Valid values are: • `Custom` • `PricingDiscoveryProcedure` • `PricingProcedure` • `ProductDiscoveryProcedure`
`Sequence`	**Type** int

Field	Details
	Properties Create, Filter, Group, Sort, Update **Description** The sequence in which the procedures are processed.
`SubSectionType`	**Type** string **Properties** Create, Filter, Group, Sort **Description** The procedure subsection added to the procedure plan definition.

ProcedurePlanVariable

Represents the setup for any adhoc user-defined variable that can be linked to a procedure plan definition record. This object is available in API version 62.0 and later.

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.

Supported SOAP API Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Supported REST API Methods

`DELETE, GET, HEAD, PATCH, POST, Query`

Fields

Field	Details
`DataType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The data type of the input procedure plan variable.
`DefaultValue`	**Type** string

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update **Description** The default value for the user-defined procedure plan variable.
`DeveloperName`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** The unique name of the procedure plan variable record. This name must begin with a letter and use only alphanumeric characters and underscores. It can't include spaces, end with an underscore, or have two consecutive underscores.
`Label`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** The label of the procedure plan variable.
`ProcedurePlanVersionId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The procedure plan version associated with the procedure plan variable record. This field is a relationship field. **Relationship Name** ProcedurePlanVersion **Relationship Type** Master-detail **Refers To** ProcedurePlanDefinitionVersion (the master object)

CHAPTER 4 Product Configurator

In this chapter ...

- Product Configurator Standard Objects
- Product Configurator Business APIs

Customize the components and attributes of a product to align with specific business requirements.

EDITIONS

Available in: Lightning Experience

Available in: **Enterprise**, **Unlimited**, and **Developer** Editions of Revenue Cloud where Product Configurator is enabled

Product Configurator Standard Objects

The Product Configurator data model provides objects and fields to manage the product configurator flow.

ExpressionSetConstraintObj
Represents the association between a Product object and the constraint model tags defined in a given constraint model. This object is available in API version 63.0 and later.

Product Configuration Flow
Specifies the many-to-many relationship between Product Classification, Product, and Flow Definition objects. The flow definition is used to configure standalone and bundled products of a specific product classification along with the product attributes, quantities, and product selling models. This object is available in API version 60.0 and later.

Product Configuration Flow Assignment
A junction entity that represents the many-to-many relationship between Product Configuration Flow, Product, and Product Classification. This object is available in API version 60.0 and later.

Product Configuration Rule
Represents the validation, inclusion, and exclusion rules for products in the context of the selling process. The selling process can be quoting, configuration, or ordering. This object is available in API version 61.0 and later.

SEE ALSO:

Object Reference for the Salesforce Platform: Overview of Salesforce Objects and Fields

SOAP API Developer Guide: Introduction to SOAP API

ExpressionSetConstraintObj

Represents the association between a Product object and the constraint model tags defined in a given constraint model. This object is available in API version 63.0 and later.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), undelete(), update(), upsert()`

Special Access Rules

This object is available in orgs where Revenue Cloud is enabled.

Fields

Field	Details
`ConstraintModelTag`	**Type** string **Properties** Create, Filter, Group, Sort, Update

Field	Details
	Description The product tag that is defined in the constraint model, for example, `Laptop`.
`ConstraintModelTagType`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update **Description** The type of the product tag that is defined in the constraint model. Possible values are: • `Port` • `Type` The default value is `Type`.
`ExpressionSetId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The expression set associated with the expression set constraint object. This field is a relationship field. **Relationship Name** ExpressionSet **Refers To** ExpressionSet
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed this record, a record related to this record, or a list view.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed this record or list view. If this value is null, the user accessed this record or list view (LastReferencedDate) but didn't view it.

Field	Details
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name of the expression set constraint.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** For internal use only. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
`ReferenceObjectId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The object associated with the expression set constraint object. This field is a polymorphic relationship field. **Relationship Name** ReferenceObject **Refers To** Product2, ProductClassification, ProductRelatedComponent

Product Configuration Flow

Specifies the many-to-many relationship between Product Classification, Product, and Flow Definition objects. The flow definition is used to configure standalone and bundled products of a specific product classification along with the product attributes, quantities, and product selling models. This object is available in API version 60.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`FlowIdentifier`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** Stores the flow API name.
`IsDefault`	**Type** Boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates the default configurator flow. The default value is `false`.
`Status`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Indicates the status of the product configuration flow. Possible values include Draft, Active, and Inactive Possible values are: • `Active` • `Draft` • `Inactive` The default value is `Draft`.

Product Configuration Flow Assignment

A junction entity that represents the many-to-many relationship between Product Configuration Flow, Product, and Product Classification. This object is available in API version 60.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
ProductClassificationId	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The product classification associated with the Product Configuration Flow. This field is a relationship field. **Relationship Name** ProductClassification **Relationship Type** Lookup **Refers To** ProductClassification
ProductConfigurationFlowId	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The Product Configuration Flow associated with the Product Classification or Product. This field is a relationship field. **Relationship Name** ProductConfigurationFlow **Relationship Type** Lookup **Refers To** ProductConfigurationFlow
ProductId	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The product associated with the Product Configuration Flow. This field is a relationship field. **Relationship Name** Product

Field	Details
	Relationship Type Lookup **Refers To** Product2

Product Configuration Rule

Represents the validation, inclusion, and exclusion rules for products in the context of the selling process. The selling process can be quoting, configuration, or ordering. This object is available in API version 61.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`ApiName`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description**
`ConfigurationRuleDefinition`	**Type** textarea **Properties** Create, Nillable, Update **Description** The configuration rule criteria and actions.
`Description`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The description of the configuration rule.
`EffectiveFromDate`	**Type** dateTime

Field	Details
	Properties Filter, Nillable, Sort **Description** The date and time from which the configuration rules comes into effect.
`EffectiveToDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date and time to which the configuration rules ceases to be in effect.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the configuration rule record was last referenced.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date the configuration rule record was last viewed.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** The name of the configuration rule.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The ID of the configuraion rule owner. This field is a polymorphic relationship field.

Field	Details
	Relationship Name Owner **Relationship Type** Lookup **Refers To** Group, User
`ProcessScope`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update **Description** The scope of the configuration rule. Possible values are: • `Bundle` • `Product` The default value is `Product`.
`RuleSubType`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update **Description** The segregation of products into subsets such that the configuration rules only apply to the products that fall under the ambit of the selected rule subtype. Possible values are: • `BundleProduct`—Bundle Product • `BundleProductClassification`—Bundle Product Classification • `Product` • `ProductClassification`—Product Classification The default value is `Product`.
`RuleType`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update **Description** Indicates the industry vertical or the feature of the industry vertical that's using the configuration rule. Possible values are:

Field	Details
	• `Configurator` • `Promotions` The default value is `Configurator`.
`Sequence`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** Indicates the order for executing the configuration rule. Rules with lower numbers run first when multiple rules are triggered at once.
`Status`	**Type** picklist **Properties** Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort **Description** The lifecycle status of the configuration rule. Possible values are: • `Active` • `Draft` • `Inactive` The default value is `Draft`.

Product Configurator Business APIs

Use the Product Configurator Business APIs to customize a product or a service according to your business-specific requirements.

Perform product configuration-related operations by using the Product Configurator Business APIs. Integrate these APIs with any front-end application to access the configurator capabilities.

This table lists the available Product Configurator resources.

Resource	Description
`/connect/cpq/configurator/actions/configure` (POST)	Retrieve and update a product's configuration from a configurator. Execute configuration rules and notify users of any violations for changes to product bundle, attributes, or product quantity within a bundle. Additionally,

Resource	Description
	get pricing details for the configured bundle.
`/connect/cpq/configurator/actions/load-instance` (POST)	Create a session for the product configuration instance using the transaction ID. Get the session ID that includes the results of actions, such as configuration rules, qualification rules, and pricing management.
`/connect/cpq/configurator/actions/set-instance` (POST)	Set a product configuration instance. This API is used in scenarios where the configuration instance is available in a different database than Salesforce and the product catalog management data is in Salesforce.
`/connect/cpq/configurator/actions/get-instance` (POST)	Fetch the JSON representation of a product configuration. Use the response to display the details of the product configuration instance on the Salesforce user interface, or save the product configuration instance to an external system.
`/connect/cpq/configurator/actions/save-instance` (POST)	Save a configuration instance after a successful product configuration.
`/connect/cpq/configurator/actions/set-product-quantity` (POST)	Set the quantity of a product through the runtime system.
`/connect/cpq/configurator/actions/add-nodes` (POST)	Add a node to the context through the runtime system without using the Salesforce user interface.
`/connect/cpq/configurator/actions/update-nodes` (POST)	Update nodes in a product configuration.
`/connect/cpq/configurator/actions/delete-nodes` (POST)	Delete nodes from a product configuration.

Resources

Learn more about the available Product Configurator API resources.

Request Bodies

Learn more about the available Product Configurator API request bodies.

Response Bodies

Learn more about the available Product Configurator API response bodies.

SEE ALSO:

Salesforce Help: Product Configurator Permissions

Resources

Learn more about the available Product Configurator API resources.

Configuration (POST)

Retrieve and update a product's configuration from a configurator. Execute configuration rules and notify users of any violations for changes to product bundle, attributes, or product quantity within a bundle. Additionally, get pricing details for the configured bundle.

Saved Configuration (GET, POST)

Save and reuse a record's configurations, and get a list of the saved configurations for a record.

Saved Configuration (DELETE, PUT)

Update or delete a record's saved configuration by using the configuration ID.

Configuration Get Instance (POST)

Fetch the JSON representation of a product configuration. Use the response to display the details of the product configuration instance on the Salesforce user interface, or save the product configuration instance to an external system.

Configuration Load Instance (POST)

Create a session for the product configuration instance using the transaction ID. Get the session ID that includes the results of actions, such as configuration rules, qualification rules, and pricing management.

Configuration Save Instance (POST)

Save a configuration instance after a successful product configuration.

Configuration Set Instance (POST)

Set a product configuration instance. This API is used in scenarios where the configuration instance is available in a different database than Salesforce and the product catalog management data is in Salesforce.

Configurator Add Nodes (POST)

Add a node to the context through the runtime system without using the Salesforce user interface.

Configurator Delete Nodes (POST)

Delete nodes from a product configuration.

Configurator Update Nodes (POST)

Update nodes in a product configuration.

Product Set Quantity (POST)

Set the quantity of a product through the runtime system.

Configuration (POST)

Retrieve and update a product's configuration from a configurator. Execute configuration rules and notify users of any violations for changes to product bundle, attributes, or product quantity within a bundle. Additionally, get pricing details for the configured bundle.

Resource

```
/connect/cpq/configurator/actions/configure
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/cpq/configurator/actions/configure
```

Available version

60.0

HTTP methods

POST

Request body for POST

JSON example

This example shows a sample to initiate a context based on a transaction ID.

```
{
    "transactionLineId": "0QLDE000000IBXw4AO",
    "transactionId": "0Q0xx0000000001GAA",
    "correlationId": "c95246d4-102c-4ecd-a263-f74ac525d1e5",
    "configuratorOptions": {
        "executePricing": true,
        "returnProductCatalogData": true
    },
    "qualificationContext": {
        "accountId": "001xx0000000001AAA",
        "contactId": "003xx00000000D7AAI"
    }
}
```

This example shows a sample to add, update, or delete a node in an existing context.

```
{
    "transactionLineId": "0QLDE000000IBXw4AO",
    "transactionId": "0Q0DE000000ISHJs81",
    "correlationId": "c95246d4-102c-4ecd-a263-f74ac525d1e5",
    "configuratorOptions": {
        "executePricing": true,
        "returnProductCatalogData": true,
        "qualifyAllProductsInTransaction": true,
        "validateProductCatalog": true,
        "validateAmendRenewCancel": true,
        "executeConfigurationRules": true,
        "addDefaultConfiguration": true
    },
    "qualificationContext": {
        "accountId": "001xx0000000001AAA",
        "contactId": "003xx00000000D7AAI"
    },
    "transactionContextId": "008d27d7-e004-4906-a949-ee7d7c323c77",
```

```
    "addedNodes": [
        {
            "path": ["0Q0DE000000ISHJs81", "sti2_id"],
            "addedObject": {
                "id": "ref_sti2_id",
                "SalesTransactionSource": "sti2_id",
                "PricebookEntry": "01uxx0000000001AAA",
                "ProductSellingModel": "0jPxx0000000001AAA",
                "businessObjectType": "QuoteLineItem",
                "Quantity": 10,
                "UnitPrice": 2.0,
                "Product": "01txx0000000001AAA"
            }
        },
        {
            "path": ["0Q0DE000000ISHJs81", "0QLDE000000IBXw4AO","ref_stir1_id"],
            "addedObject": {
                "id": "ref_stir1_id",
                "businessObjectType": "QuoteLineItemRelationship",
                "MainItem": "0QLDE000000IBXw4AO",
                "AssociatedItem": "ref_sti2_id",
                "ProductRelatedComponent": "0dSxx0000000001AAA",
                "ProductRelationshipType": "0yoxx0000000001AAA",
                "AssociatedItemPricing": "IncludedInBundlePrice"

            }
        }
    ],
    "updatedNodes": [
        {
            "path": ["0Q0DE000000ISHJs81", "0QLDE000000IBXw4AO"],
            "updatedAttributes": {
                "Quantity": 5
            }
        }
    ],
    "deletedNodes": [
        {
            "path": ["0Q0DE000000ISHJs81", "0QLDE000000IBXw4AO"]
        }
    ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`addedNodes`	Configurator Added Node Input[]	List of added context nodes that's passed to the product configurator.	Optional	60.0
`configuratorOptions`	Configurator Options Input[]	Options to pass to the configurator.	Optional	60.0

Name	Type	Description	Required or Optional	Available Version
correlationId	String	ID that's specified for traceability of logs.	Optional	60.0
deletedNodes	Configurator Deleted Node Input[]	List of deleted context nodes that's passed to the product configurator.	Optional	60.0
qualificationContext	User Context Input[]	Details such as account ID, contact ID, and context ID that are used for executing qualification rules.	Optional	60.0
transactionContextId	String	ID of the transaction context.	Optional	60.0
transactionId	String	ID of the sales transaction that's being configured such as a quote or an order.	Required	60.0
transactionLineId	String	ID of the top-level line item that's being configured.	Optional	60.0
updatedNodes	Configurator Updated Node Input[]	List of updated context nodes that's passed to the product configurator.	Optional	60.0

Response body for POST

Configuration Details

Saved Configuration (GET, POST)

Save and reuse a record's configurations, and get a list of the saved configurations for a record.

Resource

```
/connect/cpq/configurator/saved-configuration
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/cpq/configurator/saved-configuration
```

Available version

63.0

HTTP methods

GET, POST

Request parameters for GET

Parameter Name	Type	Description	Required or Optional	Available Version
referenceRecordId	String	ID of the record whose saved configurations must be retrieved.	Required	63.0

Response body for GET

Configuration List

Request body for POST

JSON example

```
{
  "data":
[illegible]
 Built-in
[illegible]

  "description": "This configuration is saved for reuse.",
  "name": "Favorite Configuration",
  "referenceRecordId": "01txx0000006iCFAAY"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`data`	String	JSON object that contains the details of the sales transaction, formatted as a string.	Optional	63.0
`description`	String	Description of the saved configuration.	Optional	63.0
`name`	String	Name of the saved configuration.	Optional	63.0
`referenceRecordId`	String	ID of the record for which the configuration must be saved.	Required	63.0

Response body for POST

Configuration Record Save

SEE ALSO:

Salesforce Help: Considerations to Save and Reuse Configurations

Saved Configuration (DELETE, PUT)

Update or delete a record's saved configuration by using the configuration ID.

Resource

```
/connect/cpq/configurator/saved-configuration/id
```

The `id` parameter is the ID of the configuration that you want to update or delete.

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/cpq/configurator/saved-configuration/5KPxx0025063GSmSAX
```

Available version

63.0

HTTP methods

DELETE, PUT

Request body for PUT

JSON example

```
{
  "data":
[illegible]
 Built-in
[illegible]

  "description": "This configuration is updated.",
  "name": "Updated Configuration"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`data`	String	JSON object that contains the details of the sales transaction, formatted as a string.	Required	63.0
`description`	String	Description of the configuration.	Required	63.0
`name`	String	Name of the configuration.	Required	63.0

Response body for PUT

Configuration Update

SEE ALSO:

Salesforce Help: Considerations to Save and Reuse Configurations

Configuration Get Instance (POST)

Fetch the JSON representation of a product configuration. Use the response to display the details of the product configuration instance on the Salesforce user interface, or save the product configuration instance to an external system.

Resource

```
/connect/cpq/configurator/actions/get-instance
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/cpq/configurator/actions/get-instance
```

Available version

60.0

HTTP methods

POST

Request body for POST

JSON example

```
{
"contextId": "008d27d7-e004-4906-a949-ee7d7c323c77"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`contextId`	String	Transaction context ID of the product configuration instance that's to be fetched.	Required	60.0

Response body for POST

Configuration Get Instance

Configuration Load Instance (POST)

Create a session for the product configuration instance using the transaction ID. Get the session ID that includes the results of actions, such as configuration rules, qualification rules, and pricing management.

Resource

```
/connect/cpq/configurator/actions/load-instance
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/cpq/configurator/actions/load-instance
```

Available version

60.0

HTTP methods

POST

Request body for POST

JSON example

```
{
    "configuratorOptions":
      {
        "addDefaultConfiguration": true,
        "executeConfigurationRules": true,
        "executePricing": true,
```

```
        "qualifyAllProductsInTransaction": true,
        "validateAmendRenewCancel": true,
        "validateProductCatalog": true
    },
    "qualificationContext": {
        "accountId": "001DU000001nHUGYA2"
    },
    "transactionId": "0Q0DU0000000XoN0AU"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`configuratorOptions`	Configurator Options Input	List of the configurator options to execute.	Optional	60.0
`contextMappingId`	String	ID of the context mapping record.	Optional	60.0
`qualificationContext`	User Context Input	Context details that are used for the qualification rules.	Optional	60.0
`transactionId`	String	Transaction ID of the header entity that's used to create a session. For example, a Quote or an Order.	Required	60.0

Response body for POST

Configuration Load Instance

Configuration Save Instance (POST)

Save a configuration instance after a successful product configuration.

Use the Configuration Save Instance API to save the changes to the source after a successful configuration. For example, save changes to the quote line item of a product, which is the source used to load the configuration.

Resource

```
/connect/cpq/configurator/actions/save-instance
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/cpq/configurator/actions/save-instance
```

Available version

60.0

HTTP methods

POST

Request body for POST

JSON example

```
{
"contextId": "008d27d7-e004-4906-a949-ee7d7c323c77"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`contextId`	String	Transaction context ID of the product configuration instance that's to be saved.	Required	60.0

Response body for POST

Configuration Save Instance

Configuration Set Instance (POST)

Set a product configuration instance. This API is used in scenarios where the configuration instance is available in a different database than Salesforce and the product catalog management data is in Salesforce.

Resource

```
/connect/cpq/configurator/actions/set-instance
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/cpq/configurator/actions/set-instance
```

Available version

60.0

HTTP methods

POST

Request body for POST

JSON example

```
{
    "configuratorOptions":
      {
        "addDefaultConfiguration": true,
        "executeConfigurationRules": true,
        "executePricing": true,
        "qualifyAllProductsInTransaction": true,
        "validateAmendRenewCancel": true,
        "validateProductCatalog": true
    },
    "qualificationContext": {
        "accountId": "001DU000001nHUGYA2"
```

```
    },
    "transactionId": "0Q0DU0000000XoN0AU"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`configuratorOptions`	Configurator Options Input	List of the configurator options to execute.	Optional	60.0
`contextMappingId`	String	ID of the context mapping record.	Required	60.0
`qualificationContext`	User Context Input	Context details that are used for the qualification rules.	Optional	60.0
`transaction`	String	Transaction JSON payload representing an object in an external system that's used to create a session.	Required	60.0

Response body for POST

Configuration Set Instance

Configurator Add Nodes (POST)

Add a node to the context through the runtime system without using the Salesforce user interface.

Resource

```
/connect/cpq/configurator/actions/add-nodes
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/cpq/configurator/actions/add-nodes
```

Available version

60.0

HTTP methods

POST

Request body for POST

JSON example

```
{
    "configuratorOptions": {
        "executePricing": true,
        "returnProductCatalogData": true,
        "qualifyAllProductsInTransaction": true,
        "validateProductCatalog": true,
        "validateAmendRenewCancel": true,
        "executeConfigurationRules": true,
        "addDefaultConfiguration": true
```

```
    },
    "qualificationContext": {
        "accountId": "001xx0000000001AAA",
        "contactId": "003xx00000000D7AAI"
    },
    "contextId": "008d27d7-e004-4906-a949-ee7d7c323c77",
    "addedNodes": [
        {
            "path": ["0Q0DE000000ISHJs81"],
            "addedObject": {
                "id": "ref_sti2_id",
                "SalesTransactionSource": "sti2_id",
                "PricebookEntry": "01uxx0000000001AAA",
                "ProductSellingModel": "0jPxx0000000001AAA",
                "businessObjectType": "QuoteLineItem",
                "Quantity": 10,
                "UnitPrice": 2.0,
                "Product": "01txx0000000001AAA"
            }
        },
        {
            "path": ["0Q0DE000000ISHJs81", "0QLDE000000IBXw4AO"],
            "addedObject": {
                "id": "ref_stir1_id",
                "businessObjectType": "QuoteLineItemRelationship",
                "MainItem": "0QLDE000000IBXw4AO",
                "AssociatedItem": "ref_sti2_id",
                "ProductRelatedComponent": "0dSxx0000000001AAA",
                "ProductRelationshipType": "0yoxx0000000001AAA",
                "AssociatedItemPricing": "IncludedInBundlePrice"

            }
        }
    ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`addedNodes`	Configurator Added Node Input[]	List of the nodes to be added.	Required	60.0
`configuratorOptions`	Configurator Options Input	List of the configuration options to execute.	Optional	60.0
`contextId`	String	ID of the context object that's being considered.	Required	60.0
`qualificationContext`	User Context Input	Context details that are used for the qualification rules.	Optional	60.0

Response body for POST

Configurator Add Nodes

Configurator Delete Nodes (POST)

Delete nodes from a product configuration.

Resource

```
/connect/cpq/configurator/actions/delete-nodes
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/cpq/configurator/actions/delete-nodes
```

Available version

60.0

HTTP methods

POST

Request body for POST

JSON example

```
{
    "configuratorOptions": {
        "executePricing": true,
        "returnProductCatalogData": true,
        "qualifyAllProductsInTransaction": true,
        "validateProductCatalog": true,
        "validateAmendRenewCancel": true,
        "executeConfigurationRules": true,
        "addDefaultConfiguration": true
    },
    "qualificationContext": {
        "accountId": "001xx0000000001AAA",
        "contactId": "003xx00000000D7AAI"
    },
    "contextId": "008d27d7-e004-4906-a949-ee7d7c323c77",
    "deletedNodes": [
        {
            "path": ["0Q0DE000000ISHJs81", "0QLDE000000IBXw4AO"]
        }
    ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`configuratorOptions`	Configurator Options Input	List of the configuration options to execute.	Optional	60.0
`contextId`	String	ID of the context object that's being considered.	Required	60.0

Name	Type	Description	Required or Optional	Available Version
deletedNodes	Configurator Deleted Node Input[]	List of the nodes to be deleted.	Required	60.0
qualificationContext	User Context Input	Context details that are used for the qualification rules.	Optional	60.0

Response body for POST

Configurator Delete Nodes

Configurator Update Nodes (POST)

Update nodes in a product configuration.

Resource

```
/connect/cpq/configurator/actions/update-nodes
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/cpq/configurator/actions/update-nodes
```

Available version

60.0

HTTP methods

POST

Request body for POST

JSON example

```
{
    "configuratorOptions": {
        "executePricing": true,
        "returnProductCatalogData": true,
        "qualifyAllProductsInTransaction": true,
        "validateProductCatalog": true,
        "validateAmendRenewCancel": true,
        "executeConfigurationRules": true,
        "addDefaultConfiguration": true
    },
    "qualificationContext": {
        "accountId": "001xx0000000001AAA",
        "contactId": "003xx00000000D7AAI"
    },
    "contextId": "008d27d7-e004-4906-a949-ee7d7c323c77",
    "updatedNodes": [
        {
            "path": ["0Q0DE000000ISHJs81", "0QLDE000000IBXw4AO"],
            "updatedAttributes": {
                "Quantity": 5
```

```
            }
        }
    ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`configuratorOptions`	Configurator Options Input	List of the configuration options to execute.	Optional	60.0
`contextId`	String	ID of the context object that's being considered.	Required	60.0
`qualificationContext`	User Context Input	Context details that are used for the qualification rules.	Optional	60.0
`updatedNodes`	Configurator Updated Node Input[]	List of the nodes to be updated.	Required	60.0

Response body for POST

Configurator Update Nodes

Product Set Quantity (POST)

Set the quantity of a product through the runtime system.

Resource

```
/connect/cpq/configurator/actions/set-product-quantity
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/cpq/configurator/actions/set-product-quantity
```

Available version

60.0

HTTP methods

POST

Request body for POST

JSON example

```
{
    "configuratorOptions": {
        "executePricing": true,
        "returnProductCatalogData": true,
        "qualifyAllProductsInTransaction": true,
        "validateProductCatalog": true,
        "validateAmendRenewCancel": true,
        "executeConfigurationRules": true,
```

```
        "addDefaultConfiguration": true
    },
    "qualificationContext": {
        "accountId": "001xx0000000001AAA",
        "contactId": "003xx00000000D7AAI"
    },
    "contextId": "008d27d7-e004-4906-a949-ee7d7c323c77",
    "quantity": 20,
    "transactionLinePath": "Quote.QuoteLineItem.Quantity"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`configuratorOptions`	Configurator Options Input	List of the configuration options to execute.	Optional	60.0
`contextId`	String	ID of the context object that's being considered.	Required	60.0
`qualificationContext`	User Context Input	Context details that are used for the qualification rules.	Optional	60.0
`quantity`	Integer	Value of the product quantity.	Required	60.0
`transactionLinePath`	String[]	Path to the line item where the update to the quantity is applied. For example, Quote.QuoteLineItem.Quantity.	Required	60.0

Response body for POST

Product Quantity Set Configurator

Request Bodies

Learn more about the available Product Configurator API request bodies.

Configuration Get Instance Input

Input representation of the request to get a product configuration instance.

Configuration Load Instance Input

Input representation of the request to load a product configuration instance.

Configuration Save Input

Input representation of the details to save a configuration.

Configuration Update Input

Input representation of the details to update a configuration.

Configuration Save Instance Input

Input representation of the request to save a product configuration instance.

Configuration Set Instance Input
Input representation of the request to set a product configuration instance.

Configurator Add Nodes Input
Input representation of the request to add nodes within a root node.

Configurator Added Node Input
Input representation of the nodes to be added to a product configuration.

Configurator Delete Nodes Input
Input representation of the request to delete nodes from a product configuration.

Configurator Deleted Node Input
Input representation of the nodes to be deleted from a product configuration.

Configurator Input
Input representation of the request to modify the product configuration.

Configurator Options Input
Input representation of the request to get the product configuration options that's passed to the configurator.

Configurator Update Nodes Input
Input representation of the request to update the nodes in a product configuration.

Configurator Updated Node Input
Input representation of the nodes to be updated in a product configuration.

Product Quantity Set Configurator Input
Input representation of the request to set the quantity of a product.

User Context Input
Input representation of the request to get the context details of a user, which are used for qualification rules.

Configuration Get Instance Input

Input representation of the request to get a product configuration instance.

JSON example

```
{
"contextId": "008d27d7-e004-4906-a949-ee7d7c323c77"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`contextId`	String	Transaction context ID of the product configuration instance that's to be fetched.	Required	60.0

Configuration Load Instance Input

Input representation of the request to load a product configuration instance.

JSON example

```
{
    "configuratorOptions":
      {
        "addDefaultConfiguration": true,
        "executeConfigurationRules": true,
        "executePricing": true,
        "qualifyAllProductsInTransaction": true,
        "validateAmendRenewCancel": true,
        "validateProductCatalog": true
    },
    "qualificationContext": {
        "accountId": "001DU000001nHUGYA2"
    },
    "transactionId": "0Q0DU0000000XoN0AU"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`configuratorOptions`	Configurator Options Input	List of the configurator options to execute.	Optional	60.0
`contextMappingId`	String	ID of the context mapping record.	Optional	60.0
`qualificationContext`	User Context Input	Context details that are used for the qualification rules.	Optional	60.0
`transactionId`	String	Transaction ID of the header entity that's used to create a session. For example, a Quote or an Order.	Required	60.0

Configuration Save Input

Input representation of the details to save a configuration.

JSON example

```
{
  "data":
[redacted]
 Built-in
[redacted]

  "description": "This configuration is saved for reuse.",
  "name": "Favorite Configuration",
```

```
  "referenceRecordId": "01txx0000006iCFAAY"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
data	String	JSON object that contains the details of the sales transaction, formatted as a string.	Optional	63.0
description	String	Description of the saved configuration.	Optional	63.0
name	String	Name of the saved configuration.	Optional	63.0
referenceRecordId	String	ID of the record for which the configuration must be saved.	Required	63.0

Configuration Update Input

Input representation of the details to update a configuration.

JSON example

```
{
  "data":
[redacted]
 Built-in
[redacted]

  "description": "This configuration is updated.",
  "name": "Updated Configuration"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
data	String	JSON object that contains the details of the sales transaction, formatted as a string.	Required	63.0
description	String	Description of the configuration.	Required	63.0
name	String	Name of the configuration.	Required	63.0

Configuration Save Instance Input

Input representation of the request to save a product configuration instance.

JSON example

```
{
"contextId": "008d27d7-e004-4906-a949-ee7d7c323c77"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
contextId	String	Transaction context ID of the product configuration instance that's to be saved.	Required	60.0

Configuration Set Instance Input

Input representation of the request to set a product configuration instance.

JSON example

```
{
    "configuratorOptions":
      {
        "addDefaultConfiguration": true,
        "executeConfigurationRules": true,
        "executePricing": true,
        "qualifyAllProductsInTransaction": true,
        "validateAmendRenewCancel": true,
        "validateProductCatalog": true
    },
    "qualificationContext": {
        "accountId": "001DU000001nHUGYA2"
    },
    "transactionId": "0Q0DU0000000XoN0AU"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
configuratorOptions	Configurator Options Input	List of the configurator options to execute.	Optional	60.0
contextMappingId	String	ID of the context mapping record.	Required	60.0
qualificationContext	User Context Input	Context details that are used for the qualification rules.	Optional	60.0

Name	Type	Description	Required or Optional	Available Version
`transaction`	String	Transaction JSON payload representing an object in an external system that's used to create a session.	Required	60.0

Configurator Add Nodes Input

Input representation of the request to add nodes within a root node.

JSON example

```
{
    "configuratorOptions": {
        "executePricing": true,
        "returnProductCatalogData": true,
        "qualifyAllProductsInTransaction": true,
        "validateProductCatalog": true,
        "validateAmendRenewCancel": true,
        "executeConfigurationRules": true,
        "addDefaultConfiguration": true
    },
    "qualificationContext": {
        "accountId": "001xx0000000001AAA",
        "contactId": "003xx00000000D7AAI"
    },
    "contextId": "008d27d7-e004-4906-a949-ee7d7c323c77",
    "addedNodes": [
        {
            "path": ["0Q0DE000000ISHJs81"],
            "addedObject": {
                "id": "ref_sti2_id",
                "SalesTransactionSource": "sti2_id",
                "PricebookEntry": "01uxx0000000001AAA",
                "ProductSellingModel": "0jPxx0000000001AAA",
                "businessObjectType": "QuoteLineItem",
                "Quantity": 10,
                "UnitPrice": 2.0,
                "Product": "01txx0000000001AAA"
            }
        },
        {
            "path": ["0Q0DE000000ISHJs81", "0QLDE000000IBXw4AO"],
            "addedObject": {
                "id": "ref_stir1_id",
                "businessObjectType": "QuoteLineItemRelationship",
                "MainItem": "0QLDE000000IBXw4AO",
                "AssociatedItem": "ref_sti2_id",
                "ProductRelatedComponent": "0dSxx0000000001AAA",
                "ProductRelationshipType": "0yoxx0000000001AAA",
                "AssociatedItemPricing": "IncludedInBundlePrice"

            }
```

```
        }
    ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`addedNodes`	Configurator Added Node Input[]	List of the nodes to be added.	Required	60.0
`configuratorOptions`	Configurator Options Input	List of the configuration options to execute.	Optional	60.0
`contextId`	String	ID of the context object that's being considered.	Required	60.0
`qualificationContext`	User Context Input	Context details that are used for the qualification rules.	Optional	60.0

Configurator Added Node Input

Input representation of the nodes to be added to a product configuration.

JSON example

```
"addedNodes": [
        {
            "path": ["0Q0DE000000ISHJs81"],
            "addedObject": {
                "id": "ref_sti2_id",
                "SalesTransactionItemSource": "sti2_id",
                "PricebookEntry": "01uxx0000000001AAA",
                "ProductSellingModel": "0jPxx0000000001AAA",
                "businessObjectType": "QuoteLineItem",
                "Quantity": 10,
                "UnitPrice": 2.0,
                "Product": "01txx0000000001AAA"
            }
        },
        {
            "path": ["0Q0DE000000ISHJs81", "0QLDE000000IBXw4AO"],
            "addedObject": {
                "id": "ref_stir1_id",
                "businessObjectType": "QuoteLineItemRelationship",
                "MainItem": "0QLDE000000IBXw4AO",
                "AssociatedItem": "ref_sti2_id",
                "ProductRelatedComponent": "0dSxx0000000001AAA",
                "ProductRelationshipType": "0yoxx0000000001AAA",
                "AssociatedItemPricing": "IncludedInBundlePrice"

            }
```

```
        }
    ]
```

Properties

Name	Type	Description	Required or Optional	Available Version
`addedObject`	Map<String, Object>	Details of the object that's being added. This property supports fields of objects from the Sales Transaction context definition, including custom objects and fields in your extended context definition.	Required	60.0
`path`	String[]	Path to the node that's being added. The path includes the unique ID of the context node in the data structure. This ID must match the ID of the sales transaction item source such as a quote line or an order line item.	Required	60.0

Configurator Delete Nodes Input

Input representation of the request to delete nodes from a product configuration.

JSON example

```
{
    "configuratorOptions": {
        "executePricing": true,
        "returnProductCatalogData": true,
        "qualifyAllProductsInTransaction": true,
        "validateProductCatalog": true,
        "validateAmendRenewCancel": true,
        "executeConfigurationRules": true,
        "addDefaultConfiguration": true
    },
    "qualificationContext": {
        "accountId": "001xx0000000001AAA",
        "contactId": "003xx00000000D7AAI"
    },
    "contextId": "008d27d7-e004-4906-a949-ee7d7c323c77",
    "deletedNodes": [
        {
            "path": ["0Q0DE000000ISHJs81", "0QLDE000000IBXw4AO"]
        }
    ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`configuratorOptions`	Configurator Options Input	List of the configuration options to execute.	Optional	60.0
`contextId`	String	ID of the context object that's being considered.	Required	60.0
`deletedNodes`	Configurator Deleted Node Input[]	List of the nodes to be deleted.	Required	60.0
`qualificationContext`	User Context Input	Context details that are used for the qualification rules.	Optional	60.0

Configurator Deleted Node Input

Input representation of the nodes to be deleted from a product configuration.

JSON example

```
    "deletedNodes": [
        {
            "path": ["0Q0DE000000ISHJs81", "0QLDE000000IBXw4AO"]
        }
    ]
```

Properties

Name	Type	Description	Required or Optional	Available Version
`path`	String[]	Path to the node that's being deleted.	Required	60.0

Configurator Input

Input representation of the request to modify the product configuration.

JSON example

This example shows a sample to initiate a context based on a transaction ID.

```
{
    "transactionLineId": "0QLDE000000IBXw4AO",
    "transactionId": "0Q0xx0000000001GAA",
    "correlationId": "c95246d4-102c-4ecd-a263-f74ac525d1e5",
    "configuratorOptions": {
        "executePricing": true,
        "returnProductCatalogData": true
    },
```

```
    "qualificationContext": {
        "accountId": "001xx0000000001AAA",
        "contactId": "003xx00000000D7AAI"
    }
}
```

This example shows a sample to add, update, or delete a node in an existing context.

```
{
    "transactionLineId": "0QLDE000000IBXw4AO",
    "transactionId": "0Q0DE000000ISHJs81",
    "correlationId": "c95246d4-102c-4ecd-a263-f74ac525d1e5",
    "configuratorOptions": {
        "executePricing": true,
        "returnProductCatalogData": true,
        "qualifyAllProductsInTransaction": true,
        "validateProductCatalog": true,
        "validateAmendRenewCancel": true,
        "executeConfigurationRules": true,
        "addDefaultConfiguration": true
    },
    "qualificationContext": {
        "accountId": "001xx0000000001AAA",
        "contactId": "003xx00000000D7AAI"
    },
    "transactionContextId": "008d27d7-e004-4906-a949-ee7d7c323c77",
    "addedNodes": [
        {
            "path": ["0Q0DE000000ISHJs81", "sti2_id"],
            "addedObject": {
                "id": "ref_sti2_id",
                "SalesTransactionSource": "sti2_id",
                "PricebookEntry": "01uxx0000000001AAA",
                "ProductSellingModel": "0jPxx0000000001AAA",
                "businessObjectType": "QuoteLineItem",
                "Quantity": 10,
                "UnitPrice": 2.0,
                "Product": "01txx0000000001AAA"
            }
        },
        {
            "path": ["0Q0DE000000ISHJs81", "0QLDE000000IBXw4AO","ref_stir1_id"],
            "addedObject": {
                "id": "ref_stir1_id",
                "businessObjectType": "QuoteLineItemRelationship",
                "MainItem": "0QLDE000000IBXw4AO",
                "AssociatedItem": "ref_sti2_id",
                "ProductRelatedComponent": "0dSxx0000000001AAA",
                "ProductRelationshipType": "0yoxx0000000001AAA",
                "AssociatedItemPricing": "IncludedInBundlePrice"

            }
        }
    ],
    "updatedNodes": [
```

```
        {
            "path": ["0Q0DE000000ISHJs81", "0QLDE000000IBXw4AO"],
            "updatedAttributes": {
                "Quantity": 5
            }
        }
    ],
    "deletedNodes": [
        {
            "path": ["0Q0DE000000ISHJs81", "0QLDE000000IBXw4AO"]
        }
    ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`addedNodes`	Configurator Added Node Input[]	List of added context nodes that's passed to the product configurator.	Optional	60.0
`configuratorOptions`	Configurator Options Input[]	Options to pass to the configurator.	Optional	60.0
`correlationId`	String	ID that's specified for traceability of logs.	Optional	60.0
`deletedNodes`	Configurator Deleted Node Input[]	List of deleted context nodes that's passed to the product configurator.	Optional	60.0
`qualificationContext`	User Context Input[]	Details such as account ID, contact ID, and context ID that are used for executing qualification rules.	Optional	60.0
`transactionContextId`	String	ID of the transaction context.	Optional	60.0
`transactionId`	String	ID of the sales transaction that's being configured such as a quote or an order.	Required	60.0
`transactionLineId`	String	ID of the top-level line item that's being configured.	Optional	60.0
`updatedNodes`	Configurator Updated Node Input[]	List of updated context nodes that's passed to the product configurator.	Optional	60.0

SEE ALSO:

Salesforce Help: Context Service

Developer Documentation: Context Service

Configurator Options Input

Input representation of the request to get the product configuration options that's passed to the configurator.

JSON example

```
"configuratorOptions":
  {
    "addDefaultConfiguration": true,
    "executeConfigurationRules": true,
    "executePricing": true,
    "qualifyAllProductsInTransaction": true,
    "validateAmendRenewCancel": true,
    "validateProductCatalog": true
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`addDefaultConfiguration`	Boolean	Indicates whether to add the default configurations (`true`) or not (`false`).	Optional	60.0
`executeConfigurationRules`	Boolean	Indicates whether to execute the configuration rules (`true`) or not (`false`).	Optional	60.0
`executePricing`	Boolean	Indicates whether to execute pricing (`true`) or not (`false`).	Optional	60.0
`pricingProcedure`	String	Name of the pricing procedure to use during the API calls to Salesforce Pricing Management.	Optional	60.0
`qualifyAllProductsInTransaction`	Boolean	Indicates whether to run the qualification rules on all the products in the context (`true`) or not (`false`).	Optional	60.0
`returnProductCatalogData`	Boolean	Indicates whether to return the product catalog data (`true`) or not (`false`). Exclude this property or specify the property value as `false` if you're using the API without the Product Configurator UI.	Optional	60.0
`validateAmendRenewCancel`	Boolean	Indicates whether to run the amend, renew, cancel-related validations (`true`) or not (`false`).	Optional	60.0
`validateProductCatalog`	Boolean	Indicates whether to run the validations against the product catalog (`true`) or not (`false`).	Optional	60.0

Configurator Update Nodes Input

Input representation of the request to update the nodes in a product configuration.

JSON example

```
{
    "configuratorOptions": {
        "executePricing": true,
        "returnProductCatalogData": true,
        "qualifyAllProductsInTransaction": true,
        "validateProductCatalog": true,
        "validateAmendRenewCancel": true,
        "executeConfigurationRules": true,
        "addDefaultConfiguration": true
    },
    "qualificationContext": {
        "accountId": "001xx0000000001AAA",
        "contactId": "003xx00000000D7AAI"
    },
    "contextId": "008d27d7-e004-4906-a949-ee7d7c323c77",
    "updatedNodes": [
        {
            "path": ["0Q0DE000000ISHJs81", "0QLDE000000IBXw4AO"],
            "updatedAttributes": {
                "Quantity": 5
            }
        }
    ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`configuratorOptions`	Configurator Options Input	List of the configuration options to execute.	Optional	60.0
`contextId`	String	ID of the context object that's being considered.	Required	60.0
`qualificationContext`	User Context Input	Context details that are used for the qualification rules.	Optional	60.0
`updatedNodes`	Configurator Updated Node Input[]	List of the nodes to be updated.	Required	60.0

Configurator Updated Node Input

Input representation of the nodes to be updated in a product configuration.

JSON example

```
    "updatedNodes": [
        {
            "path": ["0Q0DE000000ISHJs81", "0QLDE000000IBXw4AO"],
            "updatedAttributes": {
                "Quantity": 5
            }
        }
    ]
```

Properties

Name	Type	Description	Required or Optional	Available Version
`path`	String[]	Path to the node that's being updated.	Required	60.0
`updatedAttributes`	Map<String, Object>	Details of the object that's being updated. This property supports fields of objects from the Sales Transaction context definition, including custom objects and fields in your extended context definition.	Required	60.0

Product Quantity Set Configurator Input

Input representation of the request to set the quantity of a product.

JSON example

```
{
    "configuratorOptions": {
        "executePricing": true,
        "returnProductCatalogData": true,
        "qualifyAllProductsInTransaction": true,
        "validateProductCatalog": true,
        "validateAmendRenewCancel": true,
        "executeConfigurationRules": true,
        "addDefaultConfiguration": true
    },
    "qualificationContext": {
        "accountId": "001xx0000000001AAA",
        "contactId": "003xx00000000D7AAI"
    },
    "contextId": "008d27d7-e004-4906-a949-ee7d7c323c77",
    "quantity": 20,
    "transactionLinePath": "Quote.QuoteLineItem.Quantity"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`configuratorOptions`	Configurator Options Input	List of the configuration options to execute.	Optional	60.0
`contextId`	String	ID of the context object that's being considered.	Required	60.0
`qualificationContext`	User Context Input	Context details that are used for the qualification rules.	Optional	60.0
`quantity`	Integer	Value of the product quantity.	Required	60.0
`transactionLinePath`	String[]	Path to the line item where the update to the quantity is applied. For example, Quote.QuoteLineItem.Quantity.	Required	60.0

User Context Input

Input representation of the request to get the context details of a user, which are used for qualification rules.

JSON example

```
"qualificationContext": {
    "accountId": "001DU000001nHUGYA2",
    "contactId": "003xx00000000D7AAI"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`accountId`	String	ID of the account in a user context.	Optional	60.0
`contactId`	String	ID of the contact in a user context.	Optional	60.0
`contextId`	String	ID of the context that represents the created session.	Optional	60.0

Response Bodies

Learn more about the available Product Configurator API response bodies.

Configuration Details
Output representation of the product configuration details.

Configuration Get Instance
Output representation of the request to retrieve the configuration instance.

Configuration Load Instance

Output representation of the details of the context or session that are returned with a load configuration request.

Configuration Save Details

Output representation of the details of a saved configuration.

Configuration Record Save

Output representation of the details of a saved configuration.

Configuration List

Output representation of the details of the saved configuration.

Configuration Save Instance

Output representation of the response that's returned with a save configuration request.

Configuration Set Instance

Output representation of the details of the context or session that are returned with a set configuration request.

Configurator Add Nodes

Output representation of the configuration request details to add nodes.

Configurator Additional Fields

Output representation of the additional fields of a product configuration.

Configurator Attribute

Output representation of the attribute in a product configuration.

Configurator Attribute Category

Output representation of the attribute category in a product configuration.

Configurator Attribute Picklist

Output representation of the attribute picklist in a product configuration.

Configurator Attribute Picklist Value

Output representation of the values of an attribute picklist in a product configuration.

Configurator Delete Nodes

Output representation of the details of the configuration request to delete nodes.

Configurator Message

Output representation of the messages of a product configurator.

Configurator Price

Output representation of the pricing details in a product configuration.

Configurator Pricing Model

Output representation of the details of a pricing model in a product configuration.

Configurator Product Catalog

Output representation of the product catalog.

Configurator Product Classification

Output representation of the product classification in a product configuration.

Configurator Product Component Group

Output representation of the product component group in a product classification.

Configurator Product Related Component

Output representation of the product related component in a product configuration.

Configurator Product Selling Model

Output representation of the product selling model in a product configuration.

Configurator Product Selling Model Option

Output representation of the product selling model option in a product configuration.

Configurator Qualification Context

Output representation of the qualification context in a product configuration.

Configurator UI Treatment

Output representation of the details of the UI treatments of a product configurator. The details include the product configuration rule actions to override the disable or hide behavior in the UI for product options, product attributes, and attribute picklist values.

Configurator Unit Of Measure

Output representation of the details of the unit of measure record.

Configurator Update Nodes

Output representation of the configuration request details to update nodes.

Configuration Update

Output representation of the details of the updated configuration.

Error Response

Output representation of the details of the error.

Product Quantity Set Configurator

Output representation of the request details to set product quantity.

Configuration Details

Output representation of the product configuration details.

JSON example


```
{
  "catalogProducts": {
    "additionalFields": [],
    "attributeCategories": [
      {
        "attributes": [
          {
            "attributeCategoryId": "0v3xx0000000001AAA",
            "attributeNameOverride": "Load Capacity",
            "code": "CAP",
            "dataType": "NUMBER",
            "defaultValue": "1500",
          "description": "Server racks are designed to support a specific load capacity,
 commonly measured in kilograms (kg) or pounds (lbs). Typical load capacities range from
 500 kg (1102 lbs) to 1500 kg (3307 lbs) depending on the model.",
            "id": "0tjxx00000001DpAAI",
            "isCloneable": false,
            "isConfigurable": true,
            "isHidden": false,
            "isPriceImpacting": false,
            "isReadOnly": false,
            "isRequired": false,
```

```
      "label": "Load Capacity",
      "name": "Load Capacity"
    },
    {
      "attributeCategoryId": "0v3xx0000000001AAA",
      "attributeNameOverride": "Expansion Slots",
      "code": "SLOTCAP",
      "dataType": "NUMBER",
      "defaultValue": "12",
      "id": "0tjxx00000001H3AAI",
      "isCloneable": false,
      "isConfigurable": true,
      "isHidden": false,
      "isPriceImpacting": false,
      "isReadOnly": false,
      "isRequired": true,
      "label": "Expansion Slots",
      "name": "Expansion Slots"
    },
    {
      "attributeCategoryId": "0v3xx0000000001AAA",
      "attributeNameOverride": "Memory",
      "attributePicklist": {
        "id": "0v5xx0000000001AAA",
        "values": [
          {
            "code": "MEM",
            "displayValue": "25",
            "id": "0v6xx0000000001AAA",
            "isBooleanValue": false,
            "name": "25Mem",
            "sequence": 0,
            "textValue": "25"
          },
          {
            "code": "50MEM",
            "displayValue": "50",
            "id": "0v6xx000000001eAAA",
            "isBooleanValue": false,
            "name": "50Mem",
            "sequence": 1,
            "textValue": "50"
          },
          {
            "code": "100MEM",
            "displayValue": "100",
            "id": "0v6xx000000003FAAQ",
            "isBooleanValue": false,
            "name": "100Mem",
            "sequence": 2,
            "textValue": "100"
          }
        ]
      },
```

```
            "dataType": "MULTIPICKLIST",
            "defaultValue": "25",
            "id": "0tjxx00000001IfAAI",
            "isCloneable": false,
            "isConfigurable": true,
            "isHidden": false,
            "isPriceImpacting": false,
            "isReadOnly": false,
            "isRequired": true,
            "label": "Memory",
            "name": "Memory"
          }
        ],
        "code": "SPEC",
        "name": "Server Rack Specifications"
      }
    ],
    "description": "Introducing the Cisco Server Rack, a sleek and robust solution
designed to streamline your data center infrastructure. With its scalable design and
advanced cable management features, it ensures optimal performance, efficiency, and easy
 maintenance for your critical network equipment.",
    "displayUrl":
"https://www.cisco.com/content/dam/en/us/products/servers-unified-computing/ucs-c240-m4-rack-server/product-large.jpg",

    "id": "01txx0000006jkuAAA",
    "isActive": true,
    "isAssetizable": true,
    "isConfigurable": true,
    "isSoldOnlyWithOtherProds": false,
    "name": "Cisco Server Rack NX44",
    "nodeType": "bundleProduct",
    "prices": [],
    "productClassification": {
      "id": "11Bxx000002CC02EAG"
    },
    "productCode": "RACK",
    "productComponentGroups": [
      {
        "classifications": [],
        "code": "SERVICE",
        "components": [
          {
            "additionalFields": [],
            "attributeCategories": [],
            "description": "Introducing the Cisco Rack Server NX44 Service, a
comprehensive protection plan designed to safeguard your valuable data infrastructure.
 With extended coverage and rapid response times, this warranty ensures peace of mind
and uninterrupted performance for your critical business operations.",
            "id": "01txx0000006jmWAAQ",
            "isActive": true,
            "isAssetizable": true,
            "isComponentRequired": false,
            "isConfigurable": false,
            "isDefaultComponent": false,
```

```
        "isQuantityEditable": false,
        "isSoldOnlyWithOtherProds": false,
        "name": "Cisco Rack Server Service - 1 Year",
        "nodeType": "simpleProduct",
        "prices": [],
        "productClassification": {},
        "productCode": "SERVICE",
        "productComponentGroups": [],
        "productRelatedComponent": {
          "childProductId": "01txx0000006jmWAAQ",
          "childSellingModelId": "0jPxx000000004rEAA",
          "doesBundlePriceIncludeChild": true,
          "id": "0dSxx000000001dEAA",
          "isComponentRequired": false,
          "isDefaultComponent": false,
          "isQuantityEditable": false,
          "parentProductId": "01txx0000006jkuAAA",
          "parentSellingModelId": "0jPxx000000004rEAA",
          "productComponentGroupId": "0y7xx000000001dAAA",
          "productRelationshipTypeId": "0yoxx00000001IfAAI",
          "quantity": 1,
          "quantityScaleMethod": "Proportional"
        },
        "productSellingModelOptions": [
          {
            "id": "0iOxx000000009hEAA",
            "productId": "01txx0000006jmWAAQ",
            "productSellingModel": {
              "id": "0jPxx000000004rEAA",
              "name": "Termed Annually",
              "pricingTerm": 1,
              "pricingTermUnit": "Annual",
              "sellingModelType": "TermDefined",
              "status": "Active"
            },
            "productSellingModelId": "0jPxx000000004rEAA"
          },
          {
            "id": "0iOxx00000000PpEAI",
            "productId": "01txx0000006jmWAAQ",
            "productSellingModel": {
              "id": "0jPxx0000000085EAA",
              "name": "Evergreen Annually",
              "pricingTerm": 1,
              "pricingTermUnit": "Annual",
              "sellingModelType": "Evergreen",
              "status": "Active"
            },
            "productSellingModelId": "0jPxx0000000085EAA"
          }
        ]
      }
    ],
    "description": "The services available for the Cisco Server Rack NX44 product
```

```
provide comprehensive coverage and support for optimal performance and reliability,
ensuring peace of mind for your data center infrastructure.",
        "id": "0y7xx000000001dAAA",
        "maxBundleComponents": 1,
        "minBundleComponents": 0,
        "name": "Services",
        "parentProductId": "01txx0000006jkuAAA",
        "sequence": 1
      },
      {
        "classifications": [],
        "code": "WARRANTY",
        "components": [
          {
            "additionalFields": [],
            "attributeCategories": [],
            "description": "Introducing the Cisco Rack Server NX44 Warranty, a
comprehensive protection plan designed to safeguard your valuable data infrastructure.
 With extended coverage and rapid response times, this warranty ensures peace of mind
and uninterrupted performance for your critical business operations.",
            "id": "01txx0000006jjIAAQ",
            "isActive": true,
            "isAssetizable": true,
            "isComponentRequired": false,
            "isConfigurable": false,
            "isDefaultComponent": true,
            "isQuantityEditable": true,
            "isSoldOnlyWithOtherProds": false,
            "name": "Cisco Rack Server Warranty - 1 Year",
            "nodeType": "simpleProduct",
            "prices": [],
            "productClassification": {},
            "productCode": "WARRANTY",
            "productComponentGroups": [],
            "productRelatedComponent": {
              "childProductId": "01txx0000006jjIAAQ",
              "childSellingModelId": "0jPxx000000001dEAA",
              "doesBundlePriceIncludeChild": false,
              "id": "0dSxx0000000001EAA",
              "isComponentRequired": false,
              "isDefaultComponent": true,
              "isQuantityEditable": true,
              "maxQuantity": 1,
              "parentProductId": "01txx0000006jkuAAA",
              "parentSellingModelId": "0jPxx000000001dEAA",
              "productComponentGroupId": "0y7xx0000000001AAA",
              "productRelationshipTypeId": "0yoxx00000001IfAAI",
              "quantity": 1,
              "quantityScaleMethod": "Proportional",
              "sequence": 0
            },
            "productSellingModelOptions": [
              {
                "id": "0iOxx000000001dEAA",
```

```
      "productId": "01txx0000006jjIAAQ",
      "productSellingModel": {
        "id": "0jPxx000000001dEAA",
        "name": "One Time",
        "sellingModelType": "OneTime",
        "status": "Active"
      },
      "productSellingModelId": "0jPxx000000001dEAA"
    },
    {
      "id": "0iOxx00000000HlEAI",
      "productId": "01txx0000006jjIAAQ",
      "productSellingModel": {
        "id": "0jPxx000000003FEAQ",
        "name": "Termed Monthly",
        "pricingTerm": 1,
        "pricingTermUnit": "Months",
        "sellingModelType": "TermDefined",
        "status": "Active"
      },
      "productSellingModelId": "0jPxx000000003FEAQ"
    },
    {
      "id": "0iOxx00000000JNEAY",
      "productId": "01txx0000006jjIAAQ",
      "productSellingModel": {
        "id": "0jPxx000000004rEAA",
        "name": "Termed Annually",
        "pricingTerm": 1,
        "pricingTermUnit": "Annual",
        "sellingModelType": "TermDefined",
        "status": "Active"
      },
      "productSellingModelId": "0jPxx000000004rEAA"
    },
    {
      "id": "0iOxx00000000KzEAI",
      "productId": "01txx0000006jjIAAQ",
      "productSellingModel": {
        "id": "0jPxx0000000085EAA",
        "name": "Evergreen Annually",
        "pricingTerm": 1,
        "pricingTermUnit": "Annual",
        "sellingModelType": "Evergreen",
        "status": "Active"
      },
      "productSellingModelId": "0jPxx0000000085EAA"
    },
    {
      "id": "0iOxx00000000MbEAI",
      "productId": "01txx0000006jjIAAQ",
      "productSellingModel": {
        "id": "0jPxx000000006TEAQ",
        "name": "Evergreen Monthly",
```

```
                  "pricingTerm": 1,
                  "pricingTermUnit": "Months",
                  "sellingModelType": "Evergreen",
                  "status": "Active"
                },
                "productSellingModelId": "0jPxx000000006TEAQ"
              }
            ]
          }
        ],
        "description": "The warranties available for the Cisco Server Rack NX44 product
 provide comprehensive coverage and support for optimal performance and reliability,
ensuring peace of mind for your data center infrastructure.",
        "id": "0y7xx0000000001AAA",
        "maxBundleComponents": 1,
        "minBundleComponents": 0,
        "name": "Warranties",
        "parentProductId": "01txx0000006jkuAAA",
        "sequence": 0
      }
    ],
    "productSellingModelOptions": [
      {
        "id": "0iOxx000000003FEAQ",
        "productId": "01txx0000006jkuAAA",
        "productSellingModel": {
          "id": "0jPxx000000001dEAA",
          "name": "One Time",
          "sellingModelType": "OneTime",
          "status": "Active"
        },
        "productSellingModelId": "0jPxx000000001dEAA"
      },
      {
        "id": "0iOxx000000004rEAA",
        "productId": "01txx0000006jkuAAA",
        "productSellingModel": {
          "id": "0jPxx000000003FEAQ",
          "name": "Termed Monthly",
          "pricingTerm": 1,
          "pricingTermUnit": "Months",
          "sellingModelType": "TermDefined",
          "status": "Active"
        },
        "productSellingModelId": "0jPxx000000003FEAQ"
      },
      {
        "id": "0iOxx000000006TEAQ",
        "productId": "01txx0000006jkuAAA",
        "productSellingModel": {
          "id": "0jPxx000000004rEAA",
          "name": "Termed Annually",
          "pricingTerm": 1,
          "pricingTermUnit": "Annual",
```

```
          "sellingModelType": "TermDefined",
          "status": "Active"
        },
        "productSellingModelId": "0jPxx000000004rEAA"
      }
    ],
    "productType": "Bundle"
  },
  "uiTreatments": [
    {
      "details": {
        "attributeId": "0tjxx0000000007AAA",
        "prcId": "0dSxx0000000007EAA",
        "stiId": "0QLxx0000004CU0GAM",
        "attributePicklistValueId": "0v6xx0000000005AAA"
      },
      "uiTreatmentScope": "Bundle",
      "uiTreatmentTarget": "Attribute_Picklist_Value",
      "uiTreatmentType": "Hide"
    },
    {
      "details": {
        "stiId": "ref_f0f2da7b_c431_482d_bf4b_599052f3a2e1"
      },
      "uiTreatmentScope": "Product",
      "uiTreatmentTarget": "Component",
      "uiTreatmentType": "Disable"
    }
  ],
  "errors": [],
  "success": true,
  "transactionContext": {
    "SalesTransaction": [
      {
        "Status": "Draft",
        "Account": "001xx000003GeIxAAK",
        "BillingCity": "San Francisco",
        "Subtotal": 152500,
        "LastPricedDate": "2023-08-22T05:55:39Z",
        "businessObjectType": "Quote",
        "TotalAmount": 152500,
        "ShippingStreet": "415 Mission St",
        "SalesTransactionItem": [
          {
            "ProrationPolicy": null,
            "Discount": null,
            "ProductSellingModel": "0jPxx000000001dEAA",
            "Product": "01txx0000006jkuAAA",
            "businessObjectType": "QuoteLineItem",
            "BasisTransactionItem": null,
            "PartnerUnitPrice": null,
            "StartingUnitPriceSource": "System",
            "ListPrice": 150000,
            "ItemTotalAdjustmentAmount": 0,
```

```
        "SalesTransactionItemSource": "0QLxx0000004CQmGAM",
        "SubscriptionTerm": null,
        "StartDate": null,
        "NetTotalPrice": 150000,
        "TotalLineAmount": 150000,
        "PeriodBoundaryStartMonth": null,
        "ListPriceTotal": 150000,
        "PartnerDiscountPercent": null,
        "id": "0QLxx0000004CQmGAM",
        "PriceWaterFall": {
          "currencyCode": "USD",
          "executionEndTimestamp": "2023-09-18T20:11:15.016Z",
          "executionId": "ruepwmHn2ZFvnQo5bjot",
          "executionStartTimestamp": "2023-09-18T20:11:14.906Z",
          "lineItemId": "0QLxx0000004CQmGAM",
          "output": {
            "NetUnitPrice": 150000,
            "Subtotal": 0
          },
          "success": true,
          "waterfall": [
            {
              "fieldToTagNameMapping": {
                "NetUnitPrice": "ItemUnitPrice",
                "AdjustmentValue": "ItemAdjustmentValue",
                "Subtotal": "ItemTotalAdjustmentAmount",
                "Quantity": "ItemQuantity",
                "LineItemId": "SalesTransactionItemSource",
                "InputUnitPrice": "ItemUnitPrice"
              },
              "inputParameters": {
                "Quantity": 1,
                "LineItemId": "0QLxx0000004CQmGAM",
                "InputUnitPrice": 150000,
                "AdjustmentType": "Amount"
              },
              "outputParameters": {
                "NetUnitPrice": 150000,
                "Subtotal": 0
              },
              "pricingElement": {
                "adjustments": [
                  {
                    "AdjustmentType": "Amount"
                  }
                ],
                "elementType": "MANUALDISCOUNT",
                "name": "ManualDiscount"
              },
              "sequence": 1
            }
          ]
        },
        "BillingFrequency": null,
```

```
      "SalesTransactionItemParent": "0Q0xx0000004CAeCAM",
      "StartingPriceTotal": 150000,
      "Quantity": 1,
      "PeriodBoundary": null,
      "SalesTransactionItemAttribute": [
        {
          "AttributeKey": "0tjxx00000001H3AAI",
          "AttributeValue": "30.0",
          "AttributePicklistValue": null,
          "IsPriceImpacting": true,
          "businessObjectType": "QuoteLineItemAttribute",
          "AttributeName": "Expansion Slots",
          "AttributeDefinitionCode": "SLOTCAP",
          "id": "0zuxx0000000001AAA",
          "SalesTransactionItemAttrParent": "0QLxx0000004CQmGAM"
        }
      ],
      "EndDate": null,
      "DiscountAmount": null,
      "PricebookEntry": "01uxx0000009154AAA",
      "PricingTermCount": 1,
      "NetUnitPrice": 150000,
      "UnitPrice": 150000,
      "StartingUnitPrice": 150000,
      "SalesTrxnItemRelationship": [
        {
          "ProductRelationshipType": "0yoxx00000001IfAAI",
          "MainItemRole": "Bundle",
          "AssociatedItem": "0QLxx0000004CQnGAM",
          "ProductRelatedComponent": "0dSxx0000000001EAA",
          "MainItem": "0QLxx0000004CQmGAM",
          "AssociatedQuantScaleMethod": "Proportional",
          "businessObjectType": "QuoteLineRelationship",
          "AssociatedItemRole": "BundleComponent",
          "SalesTrnItemRelationshipParent": "0Q0xx0000004CAeCAM",
          "id": "0yQxx000000001dEAA",
          "AssociatedItemPricing": "NotIncludedInBundlePrice"
        }
      ],
      "TotalPrice": 150000,
      "PeriodBoundaryDay": null
    },
    {
      "ProrationPolicy": null,
      "Discount": null,
      "ProductSellingModel": "0jPxx000000001dEAA",
      "Product": "01txx0000006jjIAAQ",
      "businessObjectType": "QuoteLineItem",
      "BasisTransactionItem": null,
      "PartnerUnitPrice": null,
      "StartingUnitPriceSource": "System",
      "ListPrice": 2000,
      "ItemTotalAdjustmentAmount": 0,
      "SalesTransactionItemSource": "0QLxx0000004CQnGAM",
```

```
        "SubscriptionTerm": null,
        "StartDate": null,
        "NetTotalPrice": 2000,
        "TotalLineAmount": 2000,
        "PeriodBoundaryStartMonth": null,
        "ListPriceTotal": 2000,
        "PartnerDiscountPercent": null,
        "id": "0QLxx0000004CQnGAM",
        "PriceWaterFall": {
          "currencyCode": "USD",
          "executionEndTimestamp": "023-09-18T20:11:15.016Z",
          "executionId": "ruepwmHn2ZFvnQo5bjot",
          "executionStartTimestamp": "2023-09-18T20:11:14.906Z",
          "lineItemId": "0QLxx0000004CQnGAM",
          "output": {
            "NetUnitPrice": 2000,
            "Subtotal": 0
          },
          "success": true,
          "waterfall": [
            {
              "fieldToTagNameMapping": {
                "NetUnitPrice": "ItemUnitPrice",
                "AdjustmentValue": "ItemAdjustmentValue",
                "Subtotal": "ItemTotalAdjustmentAmount",
                "Quantity": "ItemQuantity",
                "LineItemId": "SalesTransactionItemSource",
                "InputUnitPrice": "ItemUnitPrice"
              },
              "inputParameters": {
                "Quantity": 1,
                "LineItemId": "0QLxx0000004CQnGAM",
                "InputUnitPrice": 2000,
                "AdjustmentType": "Amount"
              },
              "outputParameters": {
                "NetUnitPrice": 2000,
                "Subtotal": 0
              },
              "pricingElement": {
                "adjustments": [
                  {}
                ],
                "elementType": "MANUALDISCOUNT",
                "name": "ManualDiscount"
              },
              "sequence": 1
            }
          ]
        },
        "BillingFrequency": null,
        "SalesTransactionItemParent": "0Q0xx0000004CAeCAM",
        "StartingPriceTotal": 2000,
        "Quantity": 1,
```

```
            "PeriodBoundary": null,
            "EndDate": null,
            "DiscountAmount": null,
            "PricebookEntry": "01uxx000000913SAAQ",
            "PricingTermCount": 1,
            "NetUnitPrice": 2000,
            "UnitPrice": 2000,
            "StartingUnitPrice": 2000,
            "TotalPrice": 2000,
            "PeriodBoundaryDay": null
          }
        ],
        "BillingCountry": "US",
        "BillingStreet": "415 Mission St",
        "Pricebook": "01sxx0000005uDZAAY",
        "ShippingPostalCode": "94105",
        "SalesTransactionSource": "0Q0xx0000004CAeCAM",
        "ShippingCountry": "US",
        "ShippingCity": "San Francisco",
        "ShippingState": "CA",
        "BillingPostalCode": "94105",
        "id": "0Q0xx0000004CAeCAM",
        "BillToContact": null,
        "Contract": null,
        "BillingState": "CA"
      }
    ]
  },
  "transactionContextId": "cda87acd-45ed-4913-903e-9dd33cec85a6",
  "transactionContextMappingId": "11jxx0000004LwOAAU"
}
```


Property Name	Type	Description	Filter Group and Version	Available Version
`catalogProducts`	Configurator Product Catalog[]	Structure that contains the product catalog data.	Small, 61.0	61.0
`errors`	ConnectApiErrorResponse	List of errors that contains a message and an error code.	Small, 60.0	60.0
`messages`	Map<String, Configurator Message>>	Messages from the validation, Business Rules Engine (BRE), or Salesforce Pricing calls.	Small, 60.0	60.0
`success`	Boolean	Indicates whether the request is successful (`true`) or not (`false`).	Small, 60.0	60.0
`transactionContext`	Map<String, Object>	Serialized JSON representation of the transaction context.	Small, 60.0	60.0
`transactionContextId`	String	ID of the transaction context.	Small, 60.0	60.0

Property Name	Type	Description	Filter Group and Version	Available Version
`transactionContextMappingId`	String	ID of the context mapping.	Small, 60.0	60.0
`transactionQualification`	Map<String, Configurator Qualification Context>	Map of the product IDs to the qualification context.	Small, 60.0	60.0
`uiTreatments`	Configurator UI Treatment[]	Details of the UI treatments that specify the product configuration rule actions to override the disable or hide behavior in the UI for product options, product attributes, and attribute picklist values.	Small, 62.0	62.0

Configuration Get Instance

Output representation of the request to retrieve the configuration instance.

Property Name	Type	Description	Filter Group and Version	Available Version
`errors`	Error Response	List of errors, which contains an error code and a message.	Small, 60.0	60.0
`success`	Boolean	Indicates whether the call was successful (`true`) not (`false`).	Small, 60.0	60.0
`transaction`	Map<String, Object>	Transaction JSON payload representing an object in an external system that's used to create a session.	Small, 60.0	60.0

Configuration Load Instance

Output representation of the details of the context or session that are returned with a load configuration request.

JSON Example

```
{
  "configuratorMessages": {},
  "configuratorUITreatments": [
    {
      "details": {
        "attributeId": "0tjxx0000000007AAA",
        "prcId": "0dSxx0000000007EAA",
        "stiId": "0QLxx0000004CU0GAM",
        "attributePicklistValueId": "0v6xx0000000005AAA"
      },
      "uiTreatmentScope": "Bundle",
      "uiTreatmentTarget": "Attribute_Picklist_Value",
```

```
      "uiTreatmentType": "Hide"
    },
    {
      "details": {
        "stiId": "ref_f0f2da7b_c431_482d_bf4b_599052f3a2e1"
      },
      "uiTreatmentScope": "Product",
      "uiTreatmentTarget": "Component",
      "uiTreatmentType": "Disable"
    }
  ],
  "contextId": "831f07b01cf0cbd2d046adf5350420f85f0611b4b1e22e183921a063857a1377",
  "errors": [],
  "productQualifications": {
    "01tDU000000EOTCYA4": {
      "isQualified": true
    }
  },
  "success": true
}
```


Property Name	Type	Description	Filter Group and Version	Available Version
`configuratorMessages`	Map<String, <Configurator Message>>	Map of the product IDs to the list of configurator messages. Configurator messages are results from any validations, Business Rules Engines (BRE) calls, or Salesforce Pricing calls.	Small, 60.0	60.0
`configuratorUITreatments`	Configurator UI Treatment[]	Details of the UI treatments that specify the product configuration rule actions to override the disable or hide behavior in the UI for product options, product attributes, and attribute picklist values.	Small, 62.0	62.0
`contextId`	String	ID of the transaction context.	Small, 60.0	60.0
`errors`	Error Response	List of errors, which contains an error code and a message.	Small, 60.0	60.0
`productQualifications`	Map<String, Configurator Qualification Context>	Map of the product IDs to the execution results from qualification rules.	Small, 60.0	60.0
`success`	Boolean	Indicates whether the call was successful (`true`) not (`false`).	Small, 60.0	60.0

Configuration Save Details

Output representation of the details of a saved configuration.

JSON example

```
{
  "savedConfigurations": [
    {
      "data":
[illegible]
 Built-in
[illegible]

[illegible]

      "description": "This configuration is for cline XYZ",
      "id": "1Nyxx0000004CFUCA2",
      "name": "Client XYZ Favorite",
      "referenceRecordId": "01txx0000006iCFAAY"
    },
    {
      "data":
[illegible]
 Built-in
[illegible]

[illegible]

      "description": "This configuration is for cline XYZ",
      "id": "1Nyxx0000004CH6CAM",
      "name": "Client XYZ Favorite",
      "referenceRecordId": "01txx0000006iCFAAY"
    }
  ]
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`data`	String	JSON object that contains the details of the sales transaction, formatted as a string.	Small, 63.0	63.0
`description`	String	Description of the saved configuration.	Small, 63.0	63.0
`id`	String	ID of the saved configuration.	Small, 63.0	63.0
`name`	String	Name of the saved configuration.	Small, 63.0	63.0
`referenceRecordId`	String	ID of the record that the saved configuration belongs to.	Small, 63.0	63.0

Configuration Record Save

Output representation of the details of a saved configuration.

JSON example

This example shows a sample when the save operation is successful.

```
{
  "errors": [],
  "id": "1Nyxx0000004CNYCA2"
}
```

This example shows a sample when the save operation has errors.

```
{
  "errors": [{
  "code": "INTERNAL_SERVER_ERROR",
  "message": "INVALID_REFERENCEOBJECTID"
}]
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
errors	Error Response	List of errors that contains a message and an error code.	Small, 63.0	63.0
id	String	ID of the configuration that's saved. This property isn't shown if the operation has errors.	Small, 63.0	63.0

Configuration List

Output representation of the details of the saved configuration.

JSON example

```
{
  "errors": [],
  "savedConfigurations": [
    {
      "data":
[redacted]
 Built-in
[redacted]

      "description": "This configuration is saved for reuse.",
      "id": "1Nyxx0000004CFUCA2",
      "name": "Favorite Configuration",
      "referenceRecordId": "01txx0000006iCFAAY"
    },
    {
      "data":
[redacted]
 Built-in
```

```
[redacted]

      "description": "This configuration is saved for reuse.",
      "id": "1Nyxx0000004CH6CAM",
      "name": "Configuration 2",
      "referenceRecordId": "01txx0000006iCFAAY"
    }
  ],
  "success": true
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`errors`	Error Response	List of errors that contains a message and an error code.	Small, 63.0	63.0
`savedConfigurations`	Configuration Save Details	Configuration details associated with the `referenceRecordID` request parameter.	Small, 63.0	63.0
`success`	Boolean	Indicates whether the operation is successful (`true`) or not (`false`).	Small, 63.0	63.0

Configuration Save Instance

Output representation of the response that's returned with a save configuration request.

Property Name	Type	Description	Filter Group and Version	Available Version
`errors`	Error Response	List of errors, which contains an error code and a message.	Small, 60.0	60.0
`success`	Boolean	Indicates whether the call was successful (`true`) not (`false`).	Small, 60.0	60.0

Configuration Set Instance

Output representation of the details of the context or session that are returned with a set configuration request.

JSON Example

```
{
  "configuratorMessages": {},
  "configuratorUITreatments": [
    {
      "details": {
        "attributeId": "0tjxx0000000007AAA",
        "prcId": "0dSxx0000000007EAA",
```

```
        "stiId": "0QLxx0000004CU0GAM",
        "attributePicklistValueId": "0v6xx0000000005AAA"
      },
      "uiTreatmentScope": "Bundle",
      "uiTreatmentTarget": "Attribute_Picklist_Value",
      "uiTreatmentType": "Hide"
    },
    {
      "details": {
        "stiId": "ref_f0f2da7b_c431_482d_bf4b_599052f3a2e1"
      },
      "uiTreatmentScope": "Product",
      "uiTreatmentTarget": "Component",
      "uiTreatmentType": "Disable"
    }
  ],
  "contextId": "831f07b01cf0cbd2d046adf5350420f85f0611b4b1e22e183921a063857a1377",
  "errors": [],
  "productQualifications": {
    "01tDU000000EOTCYA4": {
      "isQualified": true
    }
  },
  "success": true
}
```


Property Name	Type	Description	Filter Group and Version	Available Version
`configuratorMessages`	Map<String, <Configurator Message>>	Map of the product IDs to the list of configurator messages. Configurator messages are results from any validations, Business Rules Engines (BRE) calls, or Salesforce Pricing calls.	Small, 60.0	60.0
`configuratorUITreatments`	Configurator UI Treatment[]	Details of the UI treatments that specify the product configuration rule actions to override the disable or hide behavior in the UI for product options, product attributes, and attribute picklist values.	Small, 62.0	62.0
`contextId`	String	ID of the transaction context.	Small, 60.0	60.0
`errors`	Error Response	List of errors, which contains an error code and a message.	Small, 60.0	60.0
`productQualifications`	Map<String, Configurator Qualification Context>	Map of the product IDs to the execution results from qualification rules.	Small, 60.0	60.0
`success`	Boolean	Indicates whether the call was successful (`true`) not (`false`).	Small, 60.0	60.0

Configurator Add Nodes

Output representation of the configuration request details to add nodes.

JSON Example

```
{
  "configuratorMessages": {},
  "configuratorUITreatments": [
    {
      "details": {
        "attributeId": "0tjxx0000000007AAA",
        "prcId": "0dSxx0000000007EAA",
        "stiId": "0QLxx0000004CU0GAM",
        "attributePicklistValueId": "0v6xx0000000005AAA"
      },
      "uiTreatmentScope": "Bundle",
      "uiTreatmentTarget": "Attribute_Picklist_Value",
      "uiTreatmentType": "Hide"
    },
    {
      "details": {
        "stiId": "ref_f0f2da7b_c431_482d_bf4b_599052f3a2e1"
      },
      "uiTreatmentScope": "Product",
      "uiTreatmentTarget": "Component",
      "uiTreatmentType": "Disable"
    }
  ],
  "errors": [],
  "productQualifications": {
    "01tDU000000EOTCYA4": {
      "isQualified": true
    }
  },
  "success": true
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`configuratorMessages`	Map<String, Configurator Message>	Map of the product IDs to the list of configurator messages. Configurator messages are results from any validations, Business Rules Engines (BRE) calls, or Salesforce Pricing calls.	Small, 60.0	60.0
`configuratorUITreatments`	Configurator UI Treatment[]	Details of the UI treatments that specify the product configuration rule actions to override the disable or hide behavior in the UI for product options, product attributes, and attribute picklist values.	Small, 62.0	62.0
`errors`	Error Response	List of errors, which contains an error code and a message.	Small, 60.0	60.0

Property Name	Type	Description	Filter Group and Version	Available Version
`product Qualifications`	Map<String, Configurator Qualification Context>	Map of the product IDs to the qualification rule execution results.	Small, 60.0	60.0
`success`	Boolean	Indicates whether the call was successful (`true`) not (`false`).	Small, 60.0	60.0

Configurator Additional Fields

Output representation of the additional fields of a product configuration.

Property Name	Type	Description	Filter Group and Version	Available Version
`fieldApiName`	String	API name of the field.	Small, 60.0	60.0
`fieldName`	String	Name of the field.	Small, 60.0	60.0
`value`	String	Value of the field.	Small, 60.0	60.0

Configurator Attribute

Output representation of the attribute in a product configuration.

JSON example

```
   "attributeCategories": [
     {
       "attributes": [
         {
           "attributeCategoryId": "0v3xx0000000001AAA",
           "attributeNameOverride": "Load Capacity",
           "code": "CAP",
           "dataType": "NUMBER",
           "defaultValue": "1500",
          "description": "Server racks are designed to support a specific load capacity,
commonly measured in kilograms (kg) or pounds (lbs). Typical load capacities range from
500 kg (1102 lbs) to 1500 kg (3307 lbs) depending on the model.",
           "id": "0tjxx00000001DpAAI",
           "isCloneable": false,
           "isConfigurable": true,
           "isHidden": false,
           "isPriceImpacting": false,
           "isReadOnly": false,
           "isRequired": false,
           "label": "Load Capacity",
           "name": "Load Capacity"
         },
```

```
        {
          "attributeCategoryId": "0v3xx0000000001AAA",
          "attributeNameOverride": "Expansion Slots",
          "code": "SLOTCAP",
          "dataType": "NUMBER",
          "defaultValue": "12",
          "id": "0tjxx00000001H3AAI",
          "isCloneable": false,
          "isConfigurable": true,
          "isHidden": false,
          "isPriceImpacting": false,
          "isReadOnly": false,
          "isRequired": true,
          "label": "Expansion Slots",
          "name": "Expansion Slots"
        },
        {
          "attributeCategoryId": "0v3xx0000000001AAA",
          "attributeNameOverride": "Memory",
          "attributePicklist": {
            "id": "0v5xx0000000001AAA",
            "values": [
              {
                "code": "MEM",
                "displayValue": "25",
                "id": "0v6xx0000000001AAA",
                "isBooleanValue": false,
                "name": "25Mem",
                "sequence": 0,
                "textValue": "25"
              },
              {
                "code": "50MEM",
                "displayValue": "50",
                "id": "0v6xx000000001eAAA",
                "isBooleanValue": false,
                "name": "50Mem",
                "sequence": 1,
                "textValue": "50"
              },
              {
                "code": "100MEM",
                "displayValue": "100",
                "id": "0v6xx000000003FAAQ",
                "isBooleanValue": false,
                "name": "100Mem",
                "sequence": 2,
                "textValue": "100"
              }
            ]
          },
          "dataType": "MULTIPICKLIST",
          "defaultValue": "25",
          "id": "0tjxx00000001IfAAI",
```

```
            "isCloneable": false,
            "isConfigurable": true,
            "isHidden": false,
            "isPriceImpacting": false,
            "isReadOnly": false,
            "isRequired": true,
            "label": "Memory",
            "name": "Memory"
          }
        ],
        "code": "SPEC",
        "name": "Server Rack Specifications"
      }
    ],
    "description": "Introducing the Cisco Server Rack, a sleek and robust solution
designed to streamline your data center infrastructure. With its scalable design and
advanced cable management features, it ensures optimal performance, efficiency, and easy
 maintenance for your critical network equipment.",
    "displayUrl":
"https://www.cisco.com/content/dam/en/us/products/servers-unified-computing/ucs-c240-m4-rack-server/product-large.jpg",

    "id": "01txx0000006jkuAAA",
    "isActive": true,
    "isAssetizable": true,
    "isConfigurable": true,
    "isSoldOnlyWithOtherProds": false,
    "name": "Cisco Server Rack NX44",
    "nodeType": "bundleProduct",
    "prices": [],
    "productClassification": {
      "id": "11Bxx000002CC02EAG"
    },
    "productCode": "RACK",
    "productComponentGroups": [
      {
        "classifications": [],
        "code": "SERVICE",
        "components": [
          {
            "additionalFields": [],
            "attributeCategories": [],
            "description": "Introducing the Cisco Rack Server NX44 Service, a
comprehensive protection plan designed to safeguard your valuable data infrastructure.
 With extended coverage and rapid response times, this warranty ensures peace of mind
and uninterrupted performance for your critical business operations.",
            "id": "01txx0000006jmWAAQ",
            "isActive": true,
            "isAssetizable": true,
            "isComponentRequired": false,
            "isConfigurable": false,
            "isDefaultComponent": false,
            "isQuantityEditable": false,
            "isSoldOnlyWithOtherProds": false,
            "name": "Cisco Rack Server Service - 1 Year",
```

```
            "nodeType": "simpleProduct",
            "prices": [],
            "productClassification": {},
            "productCode": "SERVICE",
            "productComponentGroups": [],
            "productRelatedComponent": {
              "childProductId": "01txx0000006jmWAAQ",
              "childSellingModelId": "0jPxx000000004rEAA",
              "doesBundlePriceIncludeChild": true,
              "id": "0dSxx000000001dEAA",
              "isComponentRequired": false,
              "isDefaultComponent": false,
              "isQuantityEditable": false,
              "parentProductId": "01txx0000006jkuAAA",
              "parentSellingModelId": "0jPxx000000004rEAA",
              "productComponentGroupId": "0y7xx000000001dAAA",
              "productRelationshipTypeId": "0yoxx00000001IfAAI",
              "quantity": 1,
              "quantityScaleMethod": "Proportional"
            },
            "productSellingModelOptions": [
              {
                "id": "0iOxx000000009hEAA",
                "productId": "01txx0000006jmWAAQ",
                "productSellingModel": {
                  "id": "0jPxx000000004rEAA",
                  "name": "Termed Annually",
                  "pricingTerm": 1,
                  "pricingTermUnit": "Annual",
                  "sellingModelType": "TermDefined",
                  "status": "Active"
                },
                "productSellingModelId": "0jPxx000000004rEAA"
              },
              {
                "id": "0iOxx00000000PpEAI",
                "productId": "01txx0000006jmWAAQ",
                "productSellingModel": {
                  "id": "0jPxx0000000085EAA",
                  "name": "Evergreen Annually",
                  "pricingTerm": 1,
                  "pricingTermUnit": "Annual",
                  "sellingModelType": "Evergreen",
                  "status": "Active"
                },
                "productSellingModelId": "0jPxx0000000085EAA"
              }
            ]
          }
        ],
        "description": "The services available for the Cisco Server Rack NX44 product
provide comprehensive coverage and support for optimal performance and reliability,
ensuring peace of mind for your data center infrastructure.",
        "id": "0y7xx000000001dAAA",
```

```
        "maxBundleComponents": 1,
        "minBundleComponents": 0,
        "name": "Services",
        "parentProductId": "01txx0000006jkuAAA",
        "sequence": 1
      },
      {
        "classifications": [],
        "code": "WARRANTY",
        "components": [
          {
            "additionalFields": [],
            "attributeCategories": [],
            "description": "Introducing the Cisco Rack Server NX44 Warranty, a
comprehensive protection plan designed to safeguard your valuable data infrastructure.
 With extended coverage and rapid response times, this warranty ensures peace of mind
and uninterrupted performance for your critical business operations.",
            "id": "01txx0000006jjIAAQ",
            "isActive": true,
            "isAssetizable": true,
            "isComponentRequired": false,
            "isConfigurable": false,
            "isDefaultComponent": true,
            "isQuantityEditable": true,
            "isSoldOnlyWithOtherProds": false,
            "name": "Cisco Rack Server Warranty - 1 Year",
            "nodeType": "simpleProduct",
            "prices": [],
            "productClassification": {},
            "productCode": "WARRANTY",
            "productComponentGroups": [],
            "productRelatedComponent": {
              "childProductId": "01txx0000006jjIAAQ",
              "childSellingModelId": "0jPxx000000001dEAA",
              "doesBundlePriceIncludeChild": false,
              "id": "0dSxx0000000001EAA",
              "isComponentRequired": false,
              "isDefaultComponent": true,
              "isQuantityEditable": true,
              "maxQuantity": 1,
              "parentProductId": "01txx0000006jkuAAA",
              "parentSellingModelId": "0jPxx000000001dEAA",
              "productComponentGroupId": "0y7xx0000000001AAA",
              "productRelationshipTypeId": "0yoxx00000001IfAAI",
              "quantity": 1,
              "quantityScaleMethod": "Proportional",
              "sequence": 0
            },
            "productSellingModelOptions": [
              {
                "id": "0iOxx000000001dEAA",
                "productId": "01txx0000006jjIAAQ",
                "productSellingModel": {
                  "id": "0jPxx000000001dEAA",
```

```
        "name": "One Time",
        "sellingModelType": "OneTime",
        "status": "Active"
      },
      "productSellingModelId": "0jPxx000000001dEAA"
    },
    {
      "id": "0iOxx00000000HlEAI",
      "productId": "01txx0000006jjIAAQ",
      "productSellingModel": {
        "id": "0jPxx000000003FEAQ",
        "name": "Termed Monthly",
        "pricingTerm": 1,
        "pricingTermUnit": "Months",
        "sellingModelType": "TermDefined",
        "status": "Active"
      },
      "productSellingModelId": "0jPxx000000003FEAQ"
    },
    {
      "id": "0iOxx00000000JNEAY",
      "productId": "01txx0000006jjIAAQ",
      "productSellingModel": {
        "id": "0jPxx000000004rEAA",
        "name": "Termed Annually",
        "pricingTerm": 1,
        "pricingTermUnit": "Annual",
        "sellingModelType": "TermDefined",
        "status": "Active"
      },
      "productSellingModelId": "0jPxx000000004rEAA"
    },
    {
      "id": "0iOxx00000000KzEAI",
      "productId": "01txx0000006jjIAAQ",
      "productSellingModel": {
        "id": "0jPxx0000000085EAA",
        "name": "Evergreen Annually",
        "pricingTerm": 1,
        "pricingTermUnit": "Annual",
        "sellingModelType": "Evergreen",
        "status": "Active"
      },
      "productSellingModelId": "0jPxx0000000085EAA"
    },
    {
      "id": "0iOxx00000000MbEAI",
      "productId": "01txx0000006jjIAAQ",
      "productSellingModel": {
        "id": "0jPxx000000006TEAQ",
        "name": "Evergreen Monthly",
        "pricingTerm": 1,
        "pricingTermUnit": "Months",
        "sellingModelType": "Evergreen",
```

```
                  "status": "Active"
                },
                "productSellingModelId": "0jPxx000000006TEAQ"
              }
            ]
          }
        ],
        "description": "The warranties available for the Cisco Server Rack NX44 product
 provide comprehensive coverage and support for optimal performance and reliability,
ensuring peace of mind for your data center infrastructure.",
        "id": "0y7xx0000000001AAA",
        "maxBundleComponents": 1,
        "minBundleComponents": 0,
        "name": "Warranties",
        "parentProductId": "01txx0000006jkuAAA",
        "sequence": 0
      }
    ],
    "productSellingModelOptions": [
      {
        "id": "0iOxx000000003FEAQ",
        "productId": "01txx0000006jkuAAA",
        "productSellingModel": {
          "id": "0jPxx000000001dEAA",
          "name": "One Time",
          "sellingModelType": "OneTime",
          "status": "Active"
        },
        "productSellingModelId": "0jPxx000000001dEAA"
      },
      {
        "id": "0iOxx000000004rEAA",
        "productId": "01txx0000006jkuAAA",
        "productSellingModel": {
          "id": "0jPxx000000003FEAQ",
          "name": "Termed Monthly",
          "pricingTerm": 1,
          "pricingTermUnit": "Months",
          "sellingModelType": "TermDefined",
          "status": "Active"
        },
        "productSellingModelId": "0jPxx000000003FEAQ"
      },
      {
        "id": "0iOxx000000006TEAQ",
        "productId": "01txx0000006jkuAAA",
        "productSellingModel": {
          "id": "0jPxx000000004rEAA",
          "name": "Termed Annually",
          "pricingTerm": 1,
          "pricingTermUnit": "Annual",
          "sellingModelType": "TermDefined",
          "status": "Active"
        },
```

```
      "productSellingModelId": "0jPxx000000004rEAA"
    }
  ],
  "productType": "Bundle"
},
"errors": [],
"success": true,
"transactionContext": {
  "SalesTransaction": [
    {
      "Status": "Draft",
      "Account": "001xx000003GeIxAAK",
      "BillingCity": "San Francisco",
      "Subtotal": 152500,
      "LastPricedDate": "2023-08-22T05:55:39Z",
      "businessObjectType": "Quote",
      "TotalAmount": 152500,
      "ShippingStreet": "415 Mission St",
      "SalesTransactionItem": [
        {
          "ProrationPolicy": null,
          "Discount": null,
          "ProductSellingModel": "0jPxx000000001dEAA",
          "Product": "01txx0000006jkuAAA",
          "businessObjectType": "QuoteLineItem",
          "BasisTransactionItem": null,
          "PartnerUnitPrice": null,
          "StartingUnitPriceSource": "System",
          "ListPrice": 150000,
          "ItemTotalAdjustmentAmount": 0,
          "SalesTransactionItemSource": "0QLxx0000004CQmGAM",
          "SubscriptionTerm": null,
          "StartDate": null,
          "NetTotalPrice": 150000,
          "TotalLineAmount": 150000,
          "PeriodBoundaryStartMonth": null,
          "ListPriceTotal": 150000,
          "PartnerDiscountPercent": null,
          "id": "0QLxx0000004CQmGAM",
          "PriceWaterFall": "{
              "currencyCode":"USD",
              "executionEndTimestamp":"2023-09-18T20:11:15.016Z",
              "executionId":"ruepwmHn2ZFvnQo5bjot",
              "executionStartTimestamp":"2023-09-18T20:11:14.906Z",
              "lineItemId":"0QLxx0000004CQmGAM",
              "output":{
                  "NetUnitPrice":150000.0,
                  "Subtotal":0.0
               },
              "success":true,
              "waterfall":[
              {
              "fieldToTagNameMapping":{
              "NetUnitPrice":"ItemUnitPrice",
```

```
    "AdjustmentValue":"ItemAdjustmentValue",
    "Subtotal":"ItemTotalAdjustmentAmount",
    "Quantity":"ItemQuantity",
    "LineItemId":"SalesTransactionItemSource",
    "InputUnitPrice":"ItemUnitPrice"
    },
    "inputParameters":{
        "Quantity":1.0,
        "LineItemId":"0QLxx0000004CQmGAM",
        "InputUnitPrice":150000.0,
        "AdjustmentType":"Amount"
    },
    "outputParameters":{
        "NetUnitPrice":150000.0,
        "Subtotal":0.0
    },
"pricingElement":{
    "adjustments":[
      {
        "AdjustmentType":"Amount"
      }
  ],
"elementType":"MANUALDISCOUNT",
"name":"ManualDiscount"
},
"sequence":1
}
]
}",
"BillingFrequency": null,
"SalesTransactionItemParent": "0Q0xx0000004CAeCAM",
"StartingPriceTotal": 150000,
"Quantity": 1,
"PeriodBoundary": null,
"SalesTransactionItemAttribute": [
  {
    "AttributeKey": "0tjxx00000001H3AAI",
    "AttributeValue": "30.0",
    "AttributePicklistValue": null,
    "IsPriceImpacting": true,
    "businessObjectType": "QuoteLineItemAttribute",
    "AttributeName": "Expansion Slots",
    "AttributeDefinitionCode": "SLOTCAP",
    "id": "0zuxx0000000001AAA",
    "SalesTransactionItemAttrParent": "0QLxx0000004CQmGAM"
  }
],
"EndDate": null,
"DiscountAmount": null,
"PricebookEntry": "01uxx0000009154AAA",
"PricingTermCount": 1,
"NetUnitPrice": 150000,
"UnitPrice": 150000,
"StartingUnitPrice": 150000,
```

```
      "SalesTrxnItemRelationship": [
        {
          "ProductRelationshipType": "0yoxx00000001IfAAI",
          "MainItemRole": "Bundle",
          "AssociatedItem": "0QLxx0000004CQnGAM",
          "ProductRelatedComponent": "0dSxx0000000001EAA",
          "MainItem": "0QLxx0000004CQmGAM",
          "AssociatedQuantScaleMethod": "Proportional",
          "businessObjectType": "QuoteLineRelationship",
          "AssociatedItemRole": "BundleComponent",
          "SalesTrnItemRelationshipParent": "0Q0xx0000004CAeCAM",
          "id": "0yQxx000000001dEAA",
          "AssociatedItemPricing": "NotIncludedInBundlePrice"
        }
      ],
      "TotalPrice": 150000,
      "PeriodBoundaryDay": null
    },
    {
      "ProrationPolicy": null,
      "Discount": null,
      "ProductSellingModel": "0jPxx000000001dEAA",
      "Product": "01txx0000006jjIAAQ",
      "businessObjectType": "QuoteLineItem",
      "BasisTransactionItem": null,
      "PartnerUnitPrice": null,
      "StartingUnitPriceSource": "System",
      "ListPrice": 2000,
      "ItemTotalAdjustmentAmount": 0,
      "SalesTransactionItemSource": "0QLxx0000004CQnGAM",
      "SubscriptionTerm": null,
      "StartDate": null,
      "NetTotalPrice": 2000,
      "TotalLineAmount": 2000,
      "PeriodBoundaryStartMonth": null,
      "ListPriceTotal": 2000,
      "PartnerDiscountPercent": null,
      "id": "0QLxx0000004CQnGAM",
      "PriceWaterFall": "{
          "currencyCode":"USD",
          "executionEndTimestamp":"023-09-18T20:11:15.016Z",
          "executionId":"ruepwmHn2ZFvnQo5bjot",
          "executionStartTimestamp":"2023-09-18T20:11:14.906Z",
          "lineItemId":"0QLxx0000004CQnGAM",
          "output":{
              "NetUnitPrice":2000.0,
              "Subtotal":0.0
           },
          "success":true,
          "waterfall":[
          {
          "fieldToTagNameMapping":{
          "NetUnitPrice":"ItemUnitPrice",
          "AdjustmentValue":"ItemAdjustmentValue",
```

```
            "Subtotal":"ItemTotalAdjustmentAmount",
            "Quantity":"ItemQuantity",
            "LineItemId":"SalesTransactionItemSource",
            "InputUnitPrice":"ItemUnitPrice"
            },
            "inputParameters":{
                "Quantity":1.0,
                "LineItemId":"0QLxx0000004CQnGAM",
                "InputUnitPrice":2000.0,
                "AdjustmentType":"Amount"
            },
            "outputParameters":{
                "NetUnitPrice":2000.0,
                "Subtotal":0.0
            },
        "pricingElement":{
            "adjustments":[
              {}
          ],
        "elementType":"MANUALDISCOUNT",
        "name":"ManualDiscount"
        },
        "sequence":1
        }
        ]
        }",
        "BillingFrequency": null,
        "SalesTransactionItemParent": "0Q0xx0000004CAeCAM",
        "StartingPriceTotal": 2000,
        "Quantity": 1,
        "PeriodBoundary": null,
        "EndDate": null,
        "DiscountAmount": null,
        "PricebookEntry": "01uxx000000913SAAQ",
        "PricingTermCount": 1,
        "NetUnitPrice": 2000,
        "UnitPrice": 2000,
        "StartingUnitPrice": 2000,
        "TotalPrice": 2000,
        "PeriodBoundaryDay": null
      }
    ],
    "BillingCountry": "US",
    "BillingStreet": "415 Mission St",
    "Pricebook": "01sxx0000005uDZAAY",
    "ShippingPostalCode": "94105",
    "SalesTransactionSource": "0Q0xx0000004CAeCAM",
    "ShippingCountry": "US",
    "ShippingCity": "San Francisco",
    "ShippingState": "CA",
    "BillingPostalCode": "94105",
    "id": "0Q0xx0000004CAeCAM",
    "BillToContact": null,
    "Contract": null,
```

```
        "BillingState": "CA"
      }
    ]
```

Property Name	Type	Description	Filter Group and Version	Available Version
attributeCategoryId	String	ID of the attribute category.	Small, 60.0	60.0
attributeNameOverride	String	Name override value of the attribute.	Small, 60.0	60.0
attributePicklist	Configurator Attribute Picklist[]	Picklist values of the attribute.	Small, 60.0	60.0
code	String	Code of the attribute.	Small, 60.0	60.0
dataType	String	Data type of the attribute.	Small, 60.0	60.0
defaultHelpText	String	Default help text value of the attribute.	Small, 60.0	60.0
defaultValue	String	Default value of the attribute.	Small, 60.0	60.0
description	String	Description of the attribute.	Small, 60.0	60.0
displayType	String	Display type of the attribute.	Small, 60.0	60.0
id	String	ID of the attribute.	Small, 60.0	60.0
isCloneable	Boolean	Indicates if the attribute is cloneable (true) or not (false).	Small, 60.0	60.0
isConfigurable	Boolean	Indicates if the attribute is configurable (true) or not (false).	Small, 60.0	60.0
isEncrypted	Boolean	Indicates if the attribute is encrypted (true) or not (false).	Small, 60.0	60.0
isHidden	Boolean	Indicates if the attribute is hidden (true) or not (false).	Small, 60.0	60.0
isPriceImpacting	Boolean	Indicates if this is a price impacting attribute (true) or not (false).	Small, 60.0	60.0
isReadOnly	Boolean	Indicates if the attribute is read-only (true) or not (false).	Small, 60.0	60.0
isRequired	Boolean	Indicates if the attribute is required (true) or not (false).	Small, 60.0	60.0
label	String	Label of the attribute.	Small, 60.0	60.0
maximumValue	String	Maximum value for the product attribute.	Small, 60.0	60.0
minimumValue	String	Minimum value for the product attribute.	Small, 60.0	60.0

Property Name	Type	Description	Filter Group and Version	Available Version
name	String	Name of the attribute.	Small, 60.0	60.0
sequence	Integer	Sequence values of the attribute.	Small, 60.0	60.0
status	String	Status of the attribute.	Small, 60.0	60.0
stepValue	String	Step value for the attribute.	Small, 60.0	60.0
unitOfMeasure	Configurator Unit Of Measure[]	Details about the unit of measure associated with an attribute.	Small, 63.0	63.0
userValue	String	User value of the attribute.	Small, 60.0	60.0
valueDecoder	String	Value decoder for the attribute.	Small, 60.0	60.0
value Description	String	Value description of the attribute.	Small, 60.0	60.0

Configurator Attribute Category

Output representation of the attribute category in a product configuration.

JSON example


```
   "attributeCategories": [
     {
       "attributes": [
         {
           "attributeCategoryId": "0v3xx0000000001AAA",
           "attributeNameOverride": "Load Capacity",
           "code": "CAP",
           "dataType": "NUMBER",
           "defaultValue": "1500",
          "description": "Server racks are designed to support a specific load capacity,
commonly measured in kilograms (kg) or pounds (lbs). Typical load capacities range from
500 kg (1102 lbs) to 1500 kg (3307 lbs) depending on the model.",
           "id": "0tjxx00000001DpAAI",
           "isCloneable": false,
           "isConfigurable": true,
           "isHidden": false,
           "isPriceImpacting": false,
           "isReadOnly": false,
           "isRequired": false,
           "label": "Load Capacity",
           "name": "Load Capacity"
         },
         {
           "attributeCategoryId": "0v3xx0000000001AAA",
           "attributeNameOverride": "Expansion Slots",
           "code": "SLOTCAP",
           "dataType": "NUMBER",
```

```
      "defaultValue": "12",
      "id": "0tjxx00000001H3AAI",
      "isCloneable": false,
      "isConfigurable": true,
      "isHidden": false,
      "isPriceImpacting": false,
      "isReadOnly": false,
      "isRequired": true,
      "label": "Expansion Slots",
      "name": "Expansion Slots"
    },
    {
      "attributeCategoryId": "0v3xx0000000001AAA",
      "attributeNameOverride": "Memory",
      "attributePicklist": {
        "id": "0v5xx0000000001AAA",
        "values": [
          {
            "code": "MEM",
            "displayValue": "25",
            "id": "0v6xx0000000001AAA",
            "isBooleanValue": false,
            "name": "25Mem",
            "sequence": 0,
            "textValue": "25"
          },
          {
            "code": "50MEM",
            "displayValue": "50",
            "id": "0v6xx000000001eAAA",
            "isBooleanValue": false,
            "name": "50Mem",
            "sequence": 1,
            "textValue": "50"
          },
          {
            "code": "100MEM",
            "displayValue": "100",
            "id": "0v6xx000000003FAAQ",
            "isBooleanValue": false,
            "name": "100Mem",
            "sequence": 2,
            "textValue": "100"
          }
        ]
      },
      "dataType": "MULTIPICKLIST",
      "defaultValue": "25",
      "id": "0tjxx00000001IfAAI",
      "isCloneable": false,
      "isConfigurable": true,
      "isHidden": false,
      "isPriceImpacting": false,
      "isReadOnly": false,
```

```
            "isRequired": true,
            "label": "Memory",
            "name": "Memory"
          }
        ],
        "code": "SPEC",
        "name": "Server Rack Specifications"
      }
    ],
    "description": "Introducing the Cisco Server Rack, a sleek and robust solution
designed to streamline your data center infrastructure. With its scalable design and
advanced cable management features, it ensures optimal performance, efficiency, and easy
 maintenance for your critical network equipment.",
    "displayUrl":
"https://www.cisco.com/content/dam/en/us/products/servers-unified-computing/ucs-c240-m4-rack-server/product-large.jpg",

    "id": "01txx0000006jkuAAA",
    "isActive": true,
    "isAssetizable": true,
    "isConfigurable": true,
    "isSoldOnlyWithOtherProds": false,
    "name": "Cisco Server Rack NX44",
    "nodeType": "bundleProduct",
    "prices": [],
    "productClassification": {
      "id": "11Bxx000002CC02EAG"
    },
    "productCode": "RACK",
    "productComponentGroups": [
      {
        "classifications": [],
        "code": "SERVICE",
        "components": [
          {
            "additionalFields": [],
            "attributeCategories": [],
            "description": "Introducing the Cisco Rack Server NX44 Service, a
comprehensive protection plan designed to safeguard your valuable data infrastructure.
 With extended coverage and rapid response times, this warranty ensures peace of mind
and uninterrupted performance for your critical business operations.",
            "id": "01txx0000006jmWAAQ",
            "isActive": true,
            "isAssetizable": true,
            "isComponentRequired": false,
            "isConfigurable": false,
            "isDefaultComponent": false,
            "isQuantityEditable": false,
            "isSoldOnlyWithOtherProds": false,
            "name": "Cisco Rack Server Service - 1 Year",
            "nodeType": "simpleProduct",
            "prices": [],
            "productClassification": {},
            "productCode": "SERVICE",
            "productComponentGroups": [],
```

```
            "productRelatedComponent": {
              "childProductId": "01txx0000006jmWAAQ",
              "childSellingModelId": "0jPxx000000004rEAA",
              "doesBundlePriceIncludeChild": true,
              "id": "0dSxx000000001dEAA",
              "isComponentRequired": false,
              "isDefaultComponent": false,
              "isQuantityEditable": false,
              "parentProductId": "01txx0000006jkuAAA",
              "parentSellingModelId": "0jPxx000000004rEAA",
              "productComponentGroupId": "0y7xx000000001dAAA",
              "productRelationshipTypeId": "0yoxx00000001IfAAI",
              "quantity": 1,
              "quantityScaleMethod": "Proportional"
            },
            "productSellingModelOptions": [
              {
                "id": "0iOxx000000009hEAA",
                "productId": "01txx0000006jmWAAQ",
                "productSellingModel": {
                  "id": "0jPxx000000004rEAA",
                  "name": "Termed Annually",
                  "pricingTerm": 1,
                  "pricingTermUnit": "Annual",
                  "sellingModelType": "TermDefined",
                  "status": "Active"
                },
                "productSellingModelId": "0jPxx000000004rEAA"
              },
              {
                "id": "0iOxx00000000PpEAI",
                "productId": "01txx0000006jmWAAQ",
                "productSellingModel": {
                  "id": "0jPxx0000000085EAA",
                  "name": "Evergreen Annually",
                  "pricingTerm": 1,
                  "pricingTermUnit": "Annual",
                  "sellingModelType": "Evergreen",
                  "status": "Active"
                },
                "productSellingModelId": "0jPxx0000000085EAA"
              }
            ]
          }
        ],
        "description": "The services available for the Cisco Server Rack NX44 product
provide comprehensive coverage and support for optimal performance and reliability,
ensuring peace of mind for your data center infrastructure.",
        "id": "0y7xx000000001dAAA",
        "maxBundleComponents": 1,
        "minBundleComponents": 0,
        "name": "Services",
        "parentProductId": "01txx0000006jkuAAA",
        "sequence": 1
```

```
      },
      {
        "classifications": [],
        "code": "WARRANTY",
        "components": [
          {
            "additionalFields": [],
            "attributeCategories": [],
            "description": "Introducing the Cisco Rack Server NX44 Warranty, a
comprehensive protection plan designed to safeguard your valuable data infrastructure.
 With extended coverage and rapid response times, this warranty ensures peace of mind
and uninterrupted performance for your critical business operations.",
            "id": "01txx0000006jjIAAQ",
            "isActive": true,
            "isAssetizable": true,
            "isComponentRequired": false,
            "isConfigurable": false,
            "isDefaultComponent": true,
            "isQuantityEditable": true,
            "isSoldOnlyWithOtherProds": false,
            "name": "Cisco Rack Server Warranty - 1 Year",
            "nodeType": "simpleProduct",
            "prices": [],
            "productClassification": {},
            "productCode": "WARRANTY",
            "productComponentGroups": [],
            "productRelatedComponent": {
              "childProductId": "01txx0000006jjIAAQ",
              "childSellingModelId": "0jPxx000000001dEAA",
              "doesBundlePriceIncludeChild": false,
              "id": "0dSxx0000000001EAA",
              "isComponentRequired": false,
              "isDefaultComponent": true,
              "isQuantityEditable": true,
              "maxQuantity": 1,
              "parentProductId": "01txx0000006jkuAAA",
              "parentSellingModelId": "0jPxx000000001dEAA",
              "productComponentGroupId": "0y7xx0000000001AAA",
              "productRelationshipTypeId": "0yoxx00000001IfAAI",
              "quantity": 1,
              "quantityScaleMethod": "Proportional",
              "sequence": 0
            },
            "productSellingModelOptions": [
              {
                "id": "0iOxx000000001dEAA",
                "productId": "01txx0000006jjIAAQ",
                "productSellingModel": {
                  "id": "0jPxx000000001dEAA",
                  "name": "One Time",
                  "sellingModelType": "OneTime",
                  "status": "Active"
                },
                "productSellingModelId": "0jPxx000000001dEAA"
```

```
    },
    {
      "id": "0iOxx00000000HlEAI",
      "productId": "01txx0000006jjIAAQ",
      "productSellingModel": {
        "id": "0jPxx000000003FEAQ",
        "name": "Termed Monthly",
        "pricingTerm": 1,
        "pricingTermUnit": "Months",
        "sellingModelType": "TermDefined",
        "status": "Active"
      },
      "productSellingModelId": "0jPxx000000003FEAQ"
    },
    {
      "id": "0iOxx00000000JNEAY",
      "productId": "01txx0000006jjIAAQ",
      "productSellingModel": {
        "id": "0jPxx000000004rEAA",
        "name": "Termed Annually",
        "pricingTerm": 1,
        "pricingTermUnit": "Annual",
        "sellingModelType": "TermDefined",
        "status": "Active"
      },
      "productSellingModelId": "0jPxx000000004rEAA"
    },
    {
      "id": "0iOxx00000000KzEAI",
      "productId": "01txx0000006jjIAAQ",
      "productSellingModel": {
        "id": "0jPxx0000000085EAA",
        "name": "Evergreen Annually",
        "pricingTerm": 1,
        "pricingTermUnit": "Annual",
        "sellingModelType": "Evergreen",
        "status": "Active"
      },
      "productSellingModelId": "0jPxx0000000085EAA"
    },
    {
      "id": "0iOxx00000000MbEAI",
      "productId": "01txx0000006jjIAAQ",
      "productSellingModel": {
        "id": "0jPxx000000006TEAQ",
        "name": "Evergreen Monthly",
        "pricingTerm": 1,
        "pricingTermUnit": "Months",
        "sellingModelType": "Evergreen",
        "status": "Active"
      },
      "productSellingModelId": "0jPxx000000006TEAQ"
    }
  ]
```

```
          }
        ],
        "description": "The warranties available for the Cisco Server Rack NX44 product
 provide comprehensive coverage and support for optimal performance and reliability,
ensuring peace of mind for your data center infrastructure.",
        "id": "0y7xx0000000001AAA",
        "maxBundleComponents": 1,
        "minBundleComponents": 0,
        "name": "Warranties",
        "parentProductId": "01txx0000006jkuAAA",
        "sequence": 0
      }
    ],
    "productSellingModelOptions": [
      {
        "id": "0iOxx000000003FEAQ",
        "productId": "01txx0000006jkuAAA",
        "productSellingModel": {
          "id": "0jPxx000000001dEAA",
          "name": "One Time",
          "sellingModelType": "OneTime",
          "status": "Active"
        },
        "productSellingModelId": "0jPxx000000001dEAA"
      },
      {
        "id": "0iOxx000000004rEAA",
        "productId": "01txx0000006jkuAAA",
        "productSellingModel": {
          "id": "0jPxx000000003FEAQ",
          "name": "Termed Monthly",
          "pricingTerm": 1,
          "pricingTermUnit": "Months",
          "sellingModelType": "TermDefined",
          "status": "Active"
        },
        "productSellingModelId": "0jPxx000000003FEAQ"
      },
      {
        "id": "0iOxx000000006TEAQ",
        "productId": "01txx0000006jkuAAA",
        "productSellingModel": {
          "id": "0jPxx000000004rEAA",
          "name": "Termed Annually",
          "pricingTerm": 1,
          "pricingTermUnit": "Annual",
          "sellingModelType": "TermDefined",
          "status": "Active"
        },
        "productSellingModelId": "0jPxx000000004rEAA"
      }
    ],
    "productType": "Bundle"
  },
```

```
"errors": [],
"success": true,
"transactionContext": {
  "SalesTransaction": [
    {
      "Status": "Draft",
      "Account": "001xx000003GeIxAAK",
      "BillingCity": "San Francisco",
      "Subtotal": 152500,
      "LastPricedDate": "2023-08-22T05:55:39Z",
      "businessObjectType": "Quote",
      "TotalAmount": 152500,
      "ShippingStreet": "415 Mission St",
      "SalesTransactionItem": [
        {
          "ProrationPolicy": null,
          "Discount": null,
          "ProductSellingModel": "0jPxx000000001dEAA",
          "Product": "01txx0000006jkuAAA",
          "businessObjectType": "QuoteLineItem",
          "BasisTransactionItem": null,
          "PartnerUnitPrice": null,
          "StartingUnitPriceSource": "System",
          "ListPrice": 150000,
          "ItemTotalAdjustmentAmount": 0,
          "SalesTransactionItemSource": "0QLxx0000004CQmGAM",
          "SubscriptionTerm": null,
          "StartDate": null,
          "NetTotalPrice": 150000,
          "TotalLineAmount": 150000,
          "PeriodBoundaryStartMonth": null,
          "ListPriceTotal": 150000,
          "PartnerDiscountPercent": null,
          "id": "0QLxx0000004CQmGAM",
          "PriceWaterFall": "{
              "currencyCode":"USD",
              "executionEndTimestamp":"2023-09-18T20:11:15.016Z",
              "executionId":"ruepwmHn2ZFvnQo5bjot",
              "executionStartTimestamp":"2023-09-18T20:11:14.906Z",
              "lineItemId":"0QLxx0000004CQmGAM",
              "output":{
                  "NetUnitPrice":150000.0,
                  "Subtotal":0.0
               },
              "success":true,
              "waterfall":[
              {
              "fieldToTagNameMapping":{
              "NetUnitPrice":"ItemUnitPrice",
              "AdjustmentValue":"ItemAdjustmentValue",
              "Subtotal":"ItemTotalAdjustmentAmount",
              "Quantity":"ItemQuantity",
              "LineItemId":"SalesTransactionItemSource",
              "InputUnitPrice":"ItemUnitPrice"
```

```
            },
            "inputParameters":{
                "Quantity":1.0,
                "LineItemId":"0QLxx0000004CQmGAM",
                "InputUnitPrice":150000.0,
                "AdjustmentType":"Amount"
            },
            "outputParameters":{
                "NetUnitPrice":150000.0,
                "Subtotal":0.0
            },
        "pricingElement":{
            "adjustments":[
              {
                "AdjustmentType":"Amount"
              }
          ],
        "elementType":"MANUALDISCOUNT",
        "name":"ManualDiscount"
        },
        "sequence":1
        }
        ]
        }",
        "BillingFrequency": null,
        "SalesTransactionItemParent": "0Q0xx0000004CAeCAM",
        "StartingPriceTotal": 150000,
        "Quantity": 1,
        "PeriodBoundary": null,
        "SalesTransactionItemAttribute": [
          {
            "AttributeKey": "0tjxx00000001H3AAI",
            "AttributeValue": "30.0",
            "AttributePicklistValue": null,
            "IsPriceImpacting": true,
            "businessObjectType": "QuoteLineItemAttribute",
            "AttributeName": "Expansion Slots",
            "AttributeDefinitionCode": "SLOTCAP",
            "id": "0zuxx0000000001AAA",
            "SalesTransactionItemAttrParent": "0QLxx0000004CQmGAM"
          }
        ],
        "EndDate": null,
        "DiscountAmount": null,
        "PricebookEntry": "01uxx0000009154AAA",
        "PricingTermCount": 1,
        "NetUnitPrice": 150000,
        "UnitPrice": 150000,
        "StartingUnitPrice": 150000,
        "SalesTrxnItemRelationship": [
          {
            "ProductRelationshipType": "0yoxx00000001IfAAI",
            "MainItemRole": "Bundle",
            "AssociatedItem": "0QLxx0000004CQnGAM",
```

```
            "ProductRelatedComponent": "0dSxx0000000001EAA",
            "MainItem": "0QLxx0000004CQmGAM",
            "AssociatedQuantScaleMethod": "Proportional",
            "businessObjectType": "QuoteLineRelationship",
            "AssociatedItemRole": "BundleComponent",
            "SalesTrnItemRelationshipParent": "0Q0xx0000004CAeCAM",
            "id": "0yQxx000000001dEAA",
            "AssociatedItemPricing": "NotIncludedInBundlePrice"
        }
    ],
    "TotalPrice": 150000,
    "PeriodBoundaryDay": null
  },
  {
    "ProrationPolicy": null,
    "Discount": null,
    "ProductSellingModel": "0jPxx000000001dEAA",
    "Product": "01txx0000006jjIAAQ",
    "businessObjectType": "QuoteLineItem",
    "BasisTransactionItem": null,
    "PartnerUnitPrice": null,
    "StartingUnitPriceSource": "System",
    "ListPrice": 2000,
    "ItemTotalAdjustmentAmount": 0,
    "SalesTransactionItemSource": "0QLxx0000004CQnGAM",
    "SubscriptionTerm": null,
    "StartDate": null,
    "NetTotalPrice": 2000,
    "TotalLineAmount": 2000,
    "PeriodBoundaryStartMonth": null,
    "ListPriceTotal": 2000,
    "PartnerDiscountPercent": null,
    "id": "0QLxx0000004CQnGAM",
    "PriceWaterFall": "{
        "currencyCode":"USD",
        "executionEndTimestamp":"023-09-18T20:11:15.016Z",
        "executionId":"ruepwmHn2ZFvnQo5bjot",
        "executionStartTimestamp":"2023-09-18T20:11:14.906Z",
        "lineItemId":"0QLxx0000004CQnGAM",
        "output":{
            "NetUnitPrice":2000.0,
            "Subtotal":0.0
         },
        "success":true,
        "waterfall":[
        {
        "fieldToTagNameMapping":{
        "NetUnitPrice":"ItemUnitPrice",
        "AdjustmentValue":"ItemAdjustmentValue",
        "Subtotal":"ItemTotalAdjustmentAmount",
        "Quantity":"ItemQuantity",
        "LineItemId":"SalesTransactionItemSource",
        "InputUnitPrice":"ItemUnitPrice"
        },
```

```
                "inputParameters":{
                    "Quantity":1.0,
                    "LineItemId":"0QLxx0000004CQnGAM",
                    "InputUnitPrice":2000.0,
                    "AdjustmentType":"Amount"
                },
                "outputParameters":{
                    "NetUnitPrice":2000.0,
                    "Subtotal":0.0
                },
            "pricingElement":{
                "adjustments":[
                  {}
              ],
            "elementType":"MANUALDISCOUNT",
            "name":"ManualDiscount"
            },
            "sequence":1
            }
            ]
            }",
            "BillingFrequency": null,
            "SalesTransactionItemParent": "0Q0xx0000004CAeCAM",
            "StartingPriceTotal": 2000,
            "Quantity": 1,
            "PeriodBoundary": null,
            "EndDate": null,
            "DiscountAmount": null,
            "PricebookEntry": "01uxx000000913SAAQ",
            "PricingTermCount": 1,
            "NetUnitPrice": 2000,
            "UnitPrice": 2000,
            "StartingUnitPrice": 2000,
            "TotalPrice": 2000,
            "PeriodBoundaryDay": null
          }
        ],
        "BillingCountry": "US",
        "BillingStreet": "415 Mission St",
        "Pricebook": "01sxx0000005uDZAAY",
        "ShippingPostalCode": "94105",
        "SalesTransactionSource": "0Q0xx0000004CAeCAM",
        "ShippingCountry": "US",
        "ShippingCity": "San Francisco",
        "ShippingState": "CA",
        "BillingPostalCode": "94105",
        "id": "0Q0xx0000004CAeCAM",
        "BillToContact": null,
        "Contract": null,
        "BillingState": "CA"
      }
    ]
```

Property Name	Type	Description	Filter Group and Version	Available Version
attributes	Configurator Attribute[]	Attributes of the attribute category.	Small, 60.0	60.0
code	String	Code of the attribute category.	Small, 60.0	60.0
description	String	Description of the attribute category.	Small, 60.0	60.0
id	String	ID of the attribute category.	Small, 60.0	60.0
name	String	Name of the attribute category.	Small, 60.0	60.0
status	String	Status of the attribute category.	Small, 60.0	60.0
totalSize	Integer	Total size of the attribute category.	Small, 60.0	60.0
usageType	String	Usage type of the attribute category.	Small, 60.0	60.0

Configurator Attribute Picklist

Output representation of the attribute picklist in a product configuration.

Property Name	Type	Description	Filter Group and Version	Available Version
description	String	Description of the attribute picklist.	Small, 60.0	60.0
id	String	ID of the attribute picklist.	Small, 60.0	60.0
name	String	Name of the attribute picklist.	Small, 60.0	60.0
status	String	Status of the attribute picklist.	Small, 60.0	60.0
values	Configurator Attribute Picklist Value[]	Values of the attribute picklist.	Small, 60.0	60.0

Configurator Attribute Picklist Value

Output representation of the values of an attribute picklist in a product configuration.

Property Name	Type	Description	Filter Group and Version	Available Version
code	String	Code of the attribute value.	Small, 60.0	60.0
description	String	Description of the attribute value.	Small, 60.0	60.0
displayValue	String	Display value of the attribute value.	Small, 60.0	60.0
id	String	ID of the attribute value.	Small, 60.0	60.0

Property Name	Type	Description	Filter Group and Version	Available Version
`isBooleanValue`	Boolean	Indicates if this attribute value is a boolean value (`true`) or not (`false`).	Small, 60.0	60.0
`label`	String	Label of the attribute value.	Small, 60.0	60.0
`name`	String	Name of the attribute value.	Small, 60.0	60.0
`sequence`	Integer	Sequence of the attribute value.	Small, 60.0	60.0
`status`	String	Status of the attribute value.	Small, 60.0	60.0
`textValue`	String	Text value of the attribute value.	Small, 60.0	60.0

Configurator Delete Nodes

Output representation of the details of the configuration request to delete nodes.

JSON Example

```
{
  "configuratorMessages": {},
  "configuratorUITreatments": [
    {
      "details": {
        "attributeId": "0tjxx0000000007AAA",
        "prcId": "0dSxx0000000007EAA",
        "stiId": "0QLxx0000004CU0GAM",
        "attributePicklistValueId": "0v6xx0000000005AAA"
      },
      "uiTreatmentScope": "Bundle",
      "uiTreatmentTarget": "Attribute_Picklist_Value",
      "uiTreatmentType": "Hide"
    },
    {
      "details": {
        "stiId": "ref_f0f2da7b_c431_482d_bf4b_599052f3a2e1"
      },
      "uiTreatmentScope": "Product",
      "uiTreatmentTarget": "Component",
      "uiTreatmentType": "Disable"
    }
  ],
  "errors": [],
  "productQualifications": {
    "01tDU000000EOTCYA4": {
      "isQualified": true
    }
  },
  "success": true
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`configuratorMessages`	Map<String, Configurator Message>	Map of the product IDs to the list of configurator messages. Configurator messages are results from any validations, Business Rules Engine (BRE) calls, or Salesforce Pricing calls.	Small, 60.0	60.0
`configuratorUITreatments`	Configurator UI Treatment[]	Details of the UI treatments that specify the product configuration rule actions to override the disable or hide behavior in the UI for product options, product attributes, and attribute picklist values.	Small, 62.0	62.0
`errors`	Error Response	List of errors, which contains an error code and a message.	Small, 60.0	60.0
`productQualifications`	Map<String, Configurator Qualification Context>	Map of the product IDs to the qualification rule execution results.	Small, 60.0	60.0
`success`	Boolean	Indicates whether the call was successful (`true`) not (`false`).	Small, 60.0	60.0

Configurator Message

Output representation of the messages of a product configurator.

Property Name	Type	Description	Filter Group and Version	Available Version
`category`	String	Category or type of the error message. Valid values are: • `ArcResolutionService` • `ArcValidationService` • `BundleValidation` • `ConfigurationRules` • `Pricing`	Small, 60.0	60.0
`message`	String	Message that contains the error details.	Small, 60.0	60.0
`messageType`	String	Type of error message. Valid values are: • `Error` • `Info` • `Warning`	Small, 60.0	60.0

Property Name	Type	Description	Filter Group and Version	Available Version
primaryRecordId	String	Primary record ID that contains the error.	Small, 60.0	60.0
relatedRecordId	String	Related record ID for the error, if any.	Small, 60.0	60.0

Configurator Price

Output representation of the pricing details in a product configuration.

Property Name	Type	Description	Filter Group and Version	Available Version
currencyIsoCode	String	Currency ISO code of the price book entry.	Small, 60.0	60.0
effectiveFrom	String	Date from when the price book entry is effective.	Small, 60.0	60.0
effectiveTo	String	Date until when the price book entry is effective.	Small, 60.0	60.0
isDefault	Boolean	Indicates if this price book entry is the default pricing model (true) or not (false).	Small, 60.0	60.0
isDerived	Boolean		Small, 61.0	61.0
isSelected	Boolean	Indicates if this price book entry is selected (true) or not (false).	Small, 60.0	60.0
pricebookEntryId	String	ID of the price book entry.	Small, 60.0	60.0
pricebookId	String	Pricebook2 ID of the price book entry.	Small, 60.0	60.0
pricingModel	Configurator Pricing Model[]	Pricing model details of the price book entry.	Small, 60.0	60.0
unitPrice	Double	Unit price of the price book entry.	Small, 60.0	60.0

Configurator Pricing Model

Output representation of the details of a pricing model in a product configuration.

Property Name	Type	Description	Filter Group and Version	Available Version
frequency	String	Frequency of the pricing model.	Small, 60.0	60.0

Property Name	Type	Description	Filter Group and Version	Available Version
id	String	ID of the pricing model.	Small, 60.0	60.0
name	String	Name of the pricing model.	Small, 60.0	60.0
occurrence	Integer	Occurrence of the pricing model.	Small, 60.0	60.0
pricingModelType	String	Type of the pricing model.	Small, 60.0	60.0
unitOfMeasure	String	Unit of measure for the pricing model.	Small, 60.0	60.0

Configurator Product Catalog

Output representation of the product catalog.

JSON example


```
 "catalogProducts": {
   "additionalFields": [],
   "attributeCategories": [
     {
       "attributes": [
         {
           "attributeCategoryId": "0v3xx0000000001AAA",
           "attributeNameOverride": "Load Capacity",
           "code": "CAP",
           "dataType": "NUMBER",
           "defaultValue": "1500",
          "description": "Server racks are designed to support a specific load capacity,
commonly measured in kilograms (kg) or pounds (lbs). Typical load capacities range from
500 kg (1102 lbs) to 1500 kg (3307 lbs) depending on the model.",
           "id": "0tjxx00000001DpAAI",
           "isCloneable": false,
           "isConfigurable": true,
           "isHidden": false,
           "isPriceImpacting": false,
           "isReadOnly": false,
           "isRequired": false,
           "label": "Load Capacity",
           "name": "Load Capacity"
         },
         {
           "attributeCategoryId": "0v3xx0000000001AAA",
           "attributeNameOverride": "Expansion Slots",
           "code": "SLOTCAP",
           "dataType": "NUMBER",
           "defaultValue": "12",
           "id": "0tjxx00000001H3AAI",
           "isCloneable": false,
           "isConfigurable": true,
           "isHidden": false,
```

```
      "isPriceImpacting": false,
      "isReadOnly": false,
      "isRequired": true,
      "label": "Expansion Slots",
      "name": "Expansion Slots"
    },
    {
      "attributeCategoryId": "0v3xx0000000001AAA",
      "attributeNameOverride": "Memory",
      "attributePicklist": {
        "id": "0v5xx0000000001AAA",
        "values": [
          {
            "code": "MEM",
            "displayValue": "25",
            "id": "0v6xx0000000001AAA",
            "isBooleanValue": false,
            "name": "25Mem",
            "sequence": 0,
            "textValue": "25"
          },
          {
            "code": "50MEM",
            "displayValue": "50",
            "id": "0v6xx000000001eAAA",
            "isBooleanValue": false,
            "name": "50Mem",
            "sequence": 1,
            "textValue": "50"
          },
          {
            "code": "100MEM",
            "displayValue": "100",
            "id": "0v6xx000000003FAAQ",
            "isBooleanValue": false,
            "name": "100Mem",
            "sequence": 2,
            "textValue": "100"
          }
        ]
      },
      "dataType": "MULTIPICKLIST",
      "defaultValue": "25",
      "id": "0tjxx00000001IfAAI",
      "isCloneable": false,
      "isConfigurable": true,
      "isHidden": false,
      "isPriceImpacting": false,
      "isReadOnly": false,
      "isRequired": true,
      "label": "Memory",
      "name": "Memory"
    }
  ],
```

```
        "code": "SPEC",
        "name": "Server Rack Specifications"
      }
    ],
    "description": "Introducing the Cisco Server Rack, a sleek and robust solution
designed to streamline your data center infrastructure. With its scalable design and
advanced cable management features, it ensures optimal performance, efficiency, and easy
 maintenance for your critical network equipment.",
    "displayUrl":
"https://www.cisco.com/content/dam/en/us/products/servers-unified-computing/ucs-c240-m4-rack-server/product-large.jpg",

    "id": "01txx0000006jkuAAA",
    "isActive": true,
    "isAssetizable": true,
    "isConfigurable": true,
    "isSoldOnlyWithOtherProds": false,
    "name": "Cisco Server Rack NX44",
    "nodeType": "bundleProduct",
    "prices": [],
    "productClassification": {
      "id": "11Bxx000002CC02EAG"
    },
    "productCode": "RACK",
    "productComponentGroups": [
      {
        "classifications": [],
        "code": "SERVICE",
        "components": [
          {
            "additionalFields": [],
            "attributeCategories": [],
            "description": "Introducing the Cisco Rack Server NX44 Service, a
comprehensive protection plan designed to safeguard your valuable data infrastructure.
 With extended coverage and rapid response times, this warranty ensures peace of mind
and uninterrupted performance for your critical business operations.",
            "id": "01txx0000006jmWAAQ",
            "isActive": true,
            "isAssetizable": true,
            "isComponentRequired": false,
            "isConfigurable": false,
            "isDefaultComponent": false,
            "isQuantityEditable": false,
            "isSoldOnlyWithOtherProds": false,
            "name": "Cisco Rack Server Service - 1 Year",
            "nodeType": "simpleProduct",
            "prices": [],
            "productClassification": {},
            "productCode": "SERVICE",
            "productComponentGroups": [],
            "productRelatedComponent": {
              "childProductId": "01txx0000006jmWAAQ",
              "childSellingModelId": "0jPxx000000004rEAA",
              "doesBundlePriceIncludeChild": true,
              "id": "0dSxx000000001dEAA",
```

```
              "isComponentRequired": false,
              "isDefaultComponent": false,
              "isQuantityEditable": false,
              "parentProductId": "01txx0000006jkuAAA",
              "parentSellingModelId": "0jPxx000000004rEAA",
              "productComponentGroupId": "0y7xx000000001dAAA",
              "productRelationshipTypeId": "0yoxx00000001IfAAI",
              "quantity": 1,
              "quantityScaleMethod": "Proportional"
            },
            "productSellingModelOptions": [
              {
                "id": "0iOxx000000009hEAA",
                "productId": "01txx0000006jmWAAQ",
                "productSellingModel": {
                  "id": "0jPxx000000004rEAA",
                  "name": "Termed Annually",
                  "pricingTerm": 1,
                  "pricingTermUnit": "Annual",
                  "sellingModelType": "TermDefined",
                  "status": "Active"
                },
                "productSellingModelId": "0jPxx000000004rEAA"
              },
              {
                "id": "0iOxx00000000PpEAI",
                "productId": "01txx0000006jmWAAQ",
                "productSellingModel": {
                  "id": "0jPxx0000000085EAA",
                  "name": "Evergreen Annually",
                  "pricingTerm": 1,
                  "pricingTermUnit": "Annual",
                  "sellingModelType": "Evergreen",
                  "status": "Active"
                },
                "productSellingModelId": "0jPxx0000000085EAA"
              }
            ]
          }
        ],
        "description": "The services available for the Cisco Server Rack NX44 product
provide comprehensive coverage and support for optimal performance and reliability,
ensuring peace of mind for your data center infrastructure.",
        "id": "0y7xx000000001dAAA",
        "maxBundleComponents": 1,
        "minBundleComponents": 0,
        "name": "Services",
        "parentProductId": "01txx0000006jkuAAA",
        "sequence": 1
      },
      {
        "classifications": [],
        "code": "WARRANTY",
        "components": [
```

```
          {
            "additionalFields": [],
            "attributeCategories": [],
            "description": "Introducing the Cisco Rack Server NX44 Warranty, a
comprehensive protection plan designed to safeguard your valuable data infrastructure.
 With extended coverage and rapid response times, this warranty ensures peace of mind
and uninterrupted performance for your critical business operations.",
            "id": "01txx0000006jjIAAQ",
            "isActive": true,
            "isAssetizable": true,
            "isComponentRequired": false,
            "isConfigurable": false,
            "isDefaultComponent": true,
            "isQuantityEditable": true,
            "isSoldOnlyWithOtherProds": false,
            "name": "Cisco Rack Server Warranty - 1 Year",
            "nodeType": "simpleProduct",
            "prices": [],
            "productClassification": {},
            "productCode": "WARRANTY",
            "productComponentGroups": [],
            "productRelatedComponent": {
              "childProductId": "01txx0000006jjIAAQ",
              "childSellingModelId": "0jPxx000000001dEAA",
              "doesBundlePriceIncludeChild": false,
              "id": "0dSxx0000000001EAA",
              "isComponentRequired": false,
              "isDefaultComponent": true,
              "isQuantityEditable": true,
              "maxQuantity": 1,
              "parentProductId": "01txx0000006jkuAAA",
              "parentSellingModelId": "0jPxx000000001dEAA",
              "productComponentGroupId": "0y7xx0000000001AAA",
              "productRelationshipTypeId": "0yoxx00000001IfAAI",
              "quantity": 1,
              "quantityScaleMethod": "Proportional",
              "sequence": 0
            },
            "productSellingModelOptions": [
              {
                "id": "0iOxx000000001dEAA",
                "productId": "01txx0000006jjIAAQ",
                "productSellingModel": {
                  "id": "0jPxx000000001dEAA",
                  "name": "One Time",
                  "sellingModelType": "OneTime",
                  "status": "Active"
                },
                "productSellingModelId": "0jPxx000000001dEAA"
              },
              {
                "id": "0iOxx00000000HlEAI",
                "productId": "01txx0000006jjIAAQ",
                "productSellingModel": {
```

```
                "id": "0jPxx000000003FEAQ",
                "name": "Termed Monthly",
                "pricingTerm": 1,
                "pricingTermUnit": "Months",
                "sellingModelType": "TermDefined",
                "status": "Active"
              },
              "productSellingModelId": "0jPxx000000003FEAQ"
            },
            {
              "id": "0iOxx00000000JNEAY",
              "productId": "01txx0000006jjIAAQ",
              "productSellingModel": {
                "id": "0jPxx000000004rEAA",
                "name": "Termed Annually",
                "pricingTerm": 1,
                "pricingTermUnit": "Annual",
                "sellingModelType": "TermDefined",
                "status": "Active"
              },
              "productSellingModelId": "0jPxx000000004rEAA"
            },
            {
              "id": "0iOxx00000000KzEAI",
              "productId": "01txx0000006jjIAAQ",
              "productSellingModel": {
                "id": "0jPxx0000000085EAA",
                "name": "Evergreen Annually",
                "pricingTerm": 1,
                "pricingTermUnit": "Annual",
                "sellingModelType": "Evergreen",
                "status": "Active"
              },
              "productSellingModelId": "0jPxx0000000085EAA"
            },
            {
              "id": "0iOxx00000000MbEAI",
              "productId": "01txx0000006jjIAAQ",
              "productSellingModel": {
                "id": "0jPxx000000006TEAQ",
                "name": "Evergreen Monthly",
                "pricingTerm": 1,
                "pricingTermUnit": "Months",
                "sellingModelType": "Evergreen",
                "status": "Active"
              },
              "productSellingModelId": "0jPxx000000006TEAQ"
            }
          ]
        }
      ],
      "description": "The warranties available for the Cisco Server Rack NX44 product
 provide comprehensive coverage and support for optimal performance and reliability,
ensuring peace of mind for your data center infrastructure.",
```

```
      "id": "0y7xx0000000001AAA",
      "maxBundleComponents": 1,
      "minBundleComponents": 0,
      "name": "Warranties",
      "parentProductId": "01txx0000006jkuAAA",
      "sequence": 0
    }
  ],
  "productSellingModelOptions": [
    {
      "id": "0iOxx000000003FEAQ",
      "productId": "01txx0000006jkuAAA",
      "productSellingModel": {
        "id": "0jPxx000000001dEAA",
        "name": "One Time",
        "sellingModelType": "OneTime",
        "status": "Active"
      },
      "productSellingModelId": "0jPxx000000001dEAA"
    },
    {
      "id": "0iOxx000000004rEAA",
      "productId": "01txx0000006jkuAAA",
      "productSellingModel": {
        "id": "0jPxx000000003FEAQ",
        "name": "Termed Monthly",
        "pricingTerm": 1,
        "pricingTermUnit": "Months",
        "sellingModelType": "TermDefined",
        "status": "Active"
      },
      "productSellingModelId": "0jPxx000000003FEAQ"
    },
    {
      "id": "0iOxx000000006TEAQ",
      "productId": "01txx0000006jkuAAA",
      "productSellingModel": {
        "id": "0jPxx000000004rEAA",
        "name": "Termed Annually",
        "pricingTerm": 1,
        "pricingTermUnit": "Annual",
        "sellingModelType": "TermDefined",
        "status": "Active"
      },
      "productSellingModelId": "0jPxx000000004rEAA"
    }
  ],
  "productType": "Bundle"
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
additionalFields	Configurator Additional Fields[]	Additional fields for this product.	Small, 60.0	60.0
attributeCategories	Configurator Attribute Category[]	List of attribute categories. The categories that aren't categorized are specified in the uncategorized entry in this list.	Small, 60.0	60.0
availabilityDate	String	Availability date of this product.	Small, 60.0	60.0
description	String	Description of this product.	Small, 60.0	60.0
discontinuedDate	String	Date when this product is discontinued.	Small, 60.0	60.0
displayUrl	String	Display URL of this product.	Small, 60.0	60.0
endOfLifeDate	String	End of life date for this product.	Small, 60.0	60.0
id	String	ID of the product.	Small, 60.0	60.0
isActive	Boolean	Indicates whether this product is active (`true`) or not (`false`).	Small, 60.0	60.0
isAssetizable	Boolean	Indicates whether this product is assetizable (`true`) or not (`false`).	Small, 60.0	60.0
isComponentRequired	Boolean	Indicates whether the component is required (`true`) or not (`false`).	Small, 60.0	60.0
isConfigurable	Boolean	Indicates whether the component is configurable (`true`) or not (`false`).	Small, 60.0	60.0
isDefaultComponent	Boolean	Indicates whether the component is a default component (`true`) or not (`false`).	Small, 60.0	60.0
isQuantityEditable	Boolean	Indicates whether the quantity of the component is editable (`true`) or not (`false`).	Small, 60.0	60.0
isSoldOnlyWithOtherProds	Boolean	Indicates whether this product is sold only with other products (`true`) or not (`false`).	Small, 60.0	60.0
name	String	Name of the product.	Small, 60.0	60.0
nodeType	String	Node type of the product.	Small, 60.0	60.0
prices	Configurator Price[]	List of prices from the product catalog.	Small, 60.0	60.0
productClassification	Configurator Product Classification[]	Classification details of the product.	Small, 60.0	60.0
productCode	String	Code of the product.	Small, 60.0	60.0

Property Name	Type	Description	Filter Group and Version	Available Version
`productComponentGroups`	Configurator Product Component Group[]	List of product component groups for this product.	Small, 60.0	60.0
`productRelatedComponent`	Configurator Product Related Component[]	Details of the product related component of this product.	Small, 60.0	60.0
`productSellingModelOptions`	Configurator Product Selling Model Option[]	List of product selling model options for this product.	Small, 60.0	60.0
`productType`	String	Type of product.	Small, 60.0	60.0
`qualificationContext`	Configurator Qualification Context[]	Details of the qualification context of the product.	Small, 60.0	60.0
`status`	String	Status of the product.	Small, 60.0	60.0
`unitOfMeasure`	Configurator Unit Of Measure[]	Details about the unit of measure associated with a product.	Small, 63.0	63.0

Configurator Product Classification

Output representation of the product classification in a product configuration.

JSON example

```
"productClassification": {
  "id": "11Bxx000002CC02EAG"
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`id`	String	ID of the product classification.	Small, 60.0	60.0

Configurator Product Component Group

Output representation of the product component group in a product classification.

JSON example

```
"productComponentGroups": [
  {
    "classifications": [],
    "code": "SERVICE",
    "components": [
      {
```

```
            "additionalFields": [],
            "attributeCategories": [],
            "description": "Introducing the Cisco Rack Server NX44 Service, a
comprehensive protection plan designed to safeguard your valuable data infrastructure.
 With extended coverage and rapid response times, this warranty ensures peace of mind
and uninterrupted performance for your critical business operations.",
            "id": "01txx0000006jmWAAQ",
            "isActive": true,
            "isAssetizable": true,
            "isComponentRequired": false,
            "isConfigurable": false,
            "isDefaultComponent": false,
            "isQuantityEditable": false,
            "isSoldOnlyWithOtherProds": false,
            "name": "Cisco Rack Server Service - 1 Year",
            "nodeType": "simpleProduct",
            "prices": [],
            "productClassification": {},
            "productCode": "SERVICE",
            "productComponentGroups": [],
            "productRelatedComponent": {
              "childProductId": "01txx0000006jmWAAQ",
              "childSellingModelId": "0jPxx000000004rEAA",
              "doesBundlePriceIncludeChild": true,
              "id": "0dSxx000000001dEAA",
              "isComponentRequired": false,
              "isDefaultComponent": false,
              "isQuantityEditable": false,
              "parentProductId": "01txx0000006jkuAAA",
              "parentSellingModelId": "0jPxx000000004rEAA",
              "productComponentGroupId": "0y7xx000000001dAAA",
              "productRelationshipTypeId": "0yoxx00000001IfAAI",
              "quantity": 1,
              "quantityScaleMethod": "Proportional"
            },
            "productSellingModelOptions": [
              {
                "id": "0iOxx000000009hEAA",
                "productId": "01txx0000006jmWAAQ",
                "productSellingModel": {
                  "id": "0jPxx000000004rEAA",
                  "name": "Termed Annually",
                  "pricingTerm": 1,
                  "pricingTermUnit": "Annual",
                  "sellingModelType": "TermDefined",
                  "status": "Active"
                },
                "productSellingModelId": "0jPxx000000004rEAA"
              },
              {
                "id": "0iOxx00000000PpEAI",
                "productId": "01txx0000006jmWAAQ",
                "productSellingModel": {
                  "id": "0jPxx0000000085EAA",
```

```
                  "name": "Evergreen Annually",
                  "pricingTerm": 1,
                  "pricingTermUnit": "Annual",
                  "sellingModelType": "Evergreen",
                  "status": "Active"
                },
                "productSellingModelId": "0jPxx0000000085EAA"
              }
            ]
          }
        ],
        "description": "The services available for the Cisco Server Rack NX44 product
provide comprehensive coverage and support for optimal performance and reliability,
ensuring peace of mind for your data center infrastructure.",
        "id": "0y7xx000000001dAAA",
        "maxBundleComponents": 1,
        "minBundleComponents": 0,
        "name": "Services",
        "parentProductId": "01txx0000006jkuAAA",
        "sequence": 1
      },
      {
        "classifications": [],
        "code": "WARRANTY",
        "components": [
          {
            "additionalFields": [],
            "attributeCategories": [],
            "description": "Introducing the Cisco Rack Server NX44 Warranty, a
comprehensive protection plan designed to safeguard your valuable data infrastructure.
 With extended coverage and rapid response times, this warranty ensures peace of mind
and uninterrupted performance for your critical business operations.",
            "id": "01txx0000006jjIAAQ",
            "isActive": true,
            "isAssetizable": true,
            "isComponentRequired": false,
            "isConfigurable": false,
            "isDefaultComponent": true,
            "isQuantityEditable": true,
            "isSoldOnlyWithOtherProds": false,
            "name": "Cisco Rack Server Warranty - 1 Year",
            "nodeType": "simpleProduct",
            "prices": [],
            "productClassification": {},
            "productCode": "WARRANTY",
            "productComponentGroups": [],
            "productRelatedComponent": {
              "childProductId": "01txx0000006jjIAAQ",
              "childSellingModelId": "0jPxx000000001dEAA",
              "doesBundlePriceIncludeChild": false,
              "id": "0dSxx0000000001EAA",
              "isComponentRequired": false,
              "isDefaultComponent": true,
              "isQuantityEditable": true,
```

```
      "maxQuantity": 1,
      "parentProductId": "01txx0000006jkuAAA",
      "parentSellingModelId": "0jPxx000000001dEAA",
      "productComponentGroupId": "0y7xx0000000001AAA",
      "productRelationshipTypeId": "0yoxx00000001IfAAI",
      "quantity": 1,
      "quantityScaleMethod": "Proportional",
      "sequence": 0
    },
    "productSellingModelOptions": [
      {
        "id": "0iOxx000000001dEAA",
        "productId": "01txx0000006jjIAAQ",
        "productSellingModel": {
          "id": "0jPxx000000001dEAA",
          "name": "One Time",
          "sellingModelType": "OneTime",
          "status": "Active"
        },
        "productSellingModelId": "0jPxx000000001dEAA"
      },
      {
        "id": "0iOxx00000000HlEAI",
        "productId": "01txx0000006jjIAAQ",
        "productSellingModel": {
          "id": "0jPxx000000003FEAQ",
          "name": "Termed Monthly",
          "pricingTerm": 1,
          "pricingTermUnit": "Months",
          "sellingModelType": "TermDefined",
          "status": "Active"
        },
        "productSellingModelId": "0jPxx000000003FEAQ"
      },
      {
        "id": "0iOxx00000000JNEAY",
        "productId": "01txx0000006jjIAAQ",
        "productSellingModel": {
          "id": "0jPxx000000004rEAA",
          "name": "Termed Annually",
          "pricingTerm": 1,
          "pricingTermUnit": "Annual",
          "sellingModelType": "TermDefined",
          "status": "Active"
        },
        "productSellingModelId": "0jPxx000000004rEAA"
      },
      {
        "id": "0iOxx00000000KzEAI",
        "productId": "01txx0000006jjIAAQ",
        "productSellingModel": {
          "id": "0jPxx0000000085EAA",
          "name": "Evergreen Annually",
          "pricingTerm": 1,
```

```
                "pricingTermUnit": "Annual",
                "sellingModelType": "Evergreen",
                "status": "Active"
              },
              "productSellingModelId": "0jPxx0000000085EAA"
            },
            {
              "id": "0iOxx00000000MbEAI",
              "productId": "01txx0000006jjIAAQ",
              "productSellingModel": {
                "id": "0jPxx000000006TEAQ",
                "name": "Evergreen Monthly",
                "pricingTerm": 1,
                "pricingTermUnit": "Months",
                "sellingModelType": "Evergreen",
                "status": "Active"
              },
              "productSellingModelId": "0jPxx000000006TEAQ"
            }
          ]
        }
      ],
      "description": "The warranties available for the Cisco Server Rack NX44 product
 provide comprehensive coverage and support for optimal performance and reliability,
ensuring peace of mind for your data center infrastructure.",
      "id": "0y7xx0000000001AAA",
      "maxBundleComponents": 1,
      "minBundleComponents": 0,
      "name": "Warranties",
      "parentProductId": "01txx0000006jkuAAA",
      "sequence": 0
    }
  ]
```


Property Name	Type	Description	Filter Group and Version	Available Version
classifications	Configurator Product Classification[]	List of classifications for this product component group.	Small, 60.0	60.0
code	String	Code of the product component group.	Small, 60.0	60.0
components	Configurator Product Catalog[]	Components within the product component group.	Small, 60.0	60.0
description	String	Description of the product component group.	Small, 60.0	60.0
id	String	ID of the product component group.	Small, 60.0	60.0
maxBundle Components	Integer	Maximum number of bundle components within the product component group.	Small, 60.0	60.0
minBundle Components	Integer	Minimum number of bundle components within the product component group.	Small, 60.0	60.0

Property Name	Type	Description	Filter Group and Version	Available Version
name	String	Name of the product component group.	Small, 60.0	60.0
parentProductId	String	Parent Product2 ID of the product component group.	Small, 60.0	60.0
sequence	Integer	Sequence of the product component group.	Small, 60.0	60.0

Configurator Product Related Component

Output representation of the product related component in a product configuration.

JSON example

```
"productRelatedComponent": {
  "childProductId": "01txx0000006jmWAAQ",
  "childSellingModelId": "0jPxx000000004rEAA",
  "doesBundlePriceIncludeChild": true,
  "id": "0dSxx000000001dEAA",
  "isComponentRequired": false,
  "isDefaultComponent": false,
  "isQuantityEditable": false,
  "parentProductId": "01txx0000006jkuAAA",
  "parentSellingModelId": "0jPxx000000004rEAA",
  "productComponentGroupId": "0y7xx000000001dAAA",
  "productRelationshipTypeId": "0yoxx00000001IfAAI",
  "quantity": 1,
  "quantityScaleMethod": "Proportional"
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
childProductId	String	Child Product ID	Small, 60.0	60.0
childSellingModelId	String	Child PSM Id	Small, 60.0	60.0
doesBundlePriceIncludeChild	Boolean	Does the bundle price include child products	Small, 60.0	60.0
id	String	ID of this Product Related Component	Small, 60.0	60.0
isComponentRequired	Boolean	Is Component Required	Small, 60.0	60.0
isDefaultComponent	Boolean	Is Default Component	Small, 60.0	60.0

Property Name	Type	Description	Filter Group and Version	Available Version
isQuantityEditable	Boolean	Is Quantity Editable	Small, 60.0	60.0
maxQuantity	Double	Max Quantity	Small, 60.0	60.0
minQuantity	Double	Min Quantity	Small, 60.0	60.0
parentProductId	String	Parent Product ID	Small, 60.0	60.0
parentSellingModelId	String	Parent PSM Id	Small, 60.0	60.0
productClassificationId	String	Product Classification ID	Small, 60.0	60.0
productComponentGroupId	String	Product Component Group ID	Small, 60.0	60.0
productRelationshipTypeId	String	PRT Id	Small, 60.0	60.0
quantity	Double	Default Quantity	Small, 60.0	60.0
quantityScaleMethod	String	Quantity Scale Method	Small, 60.0	60.0
sequence	Integer	Sequence	Small, 60.0	60.0

Configurator Product Selling Model

Output representation of the product selling model in a product configuration.

JSON example

```
    "productSellingModel": {
        "id": "0jPxx0000000085EAA",
        "name": "Evergreen Annually",
        "pricingTerm": 1,
        "pricingTermUnit": "Annual",
        "sellingModelType": "Evergreen",
        "status": "Active"
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
id	String	ID of the product selling model.	Small, 60.0	60.0

Property Name	Type	Description	Filter Group and Version	Available Version
name	String	Name of the product selling model.	Small, 60.0	60.0
pricingTerm	Integer	Pricing term of the product selling model.	Small, 60.0	60.0
pricingTermUnit	String	Pricing term unit of the product selling model.	Small, 60.0	60.0
sellingModelType	String	Selling model type of the product selling model.	Small, 60.0	60.0
status	String	Status of the product selling model.	Small, 60.0	60.0

Configurator Product Selling Model Option

Output representation of the product selling model option in a product configuration.

JSON example

```
"productSellingModelOptions": [
  {
    "id": "0iOxx000000009hEAA",
    "productId": "01txx0000006jmWAAQ",
    "productSellingModel": {
      "id": "0jPxx000000004rEAA",
   "name": "Termed Annually",
      "pricingTerm": 1,
      "pricingTermUnit": "Annual",
      "sellingModelType": "TermDefined",
      "status": "Active"
    },
    "productSellingModelId": "0jPxx000000004rEAA"
  },
  {
    "id": "0iOxx00000000PpEAI",
    "productId": "01txx0000006jmWAAQ",
    "productSellingModel": {
      "id": "0jPxx0000000085EAA",
      "name": "Evergreen Annually",
      "pricingTerm": 1,
      "pricingTermUnit": "Annual",
      "sellingModelType": "Evergreen",
      "status": "Active"
    },
    "productSellingModelId": "0jPxx0000000085EAA"
  }
]
```

Property Name	Type	Description	Filter Group and Version	Available Version
`id`	String	ID of the product selling model option.	Small, 60.0	60.0
`productId`	String	ID of the product that's associated with the product selling model option.	Small, 60.0	60.0
`productSellingModel`	Configurator Product Selling Model[]	Product selling model that's associated with the product selling model option.	Small, 60.0	60.0
`productSellingModelId`	String	ID of the product selling model that's associated with the product selling model option.	Small, 60.0	60.0

Configurator Qualification Context

Output representation of the qualification context in a product configuration.

Property Name	Type	Description	Filter Group and Version	Available Version
`isQualified`	Boolean	Indicates whether the product is qualified (`true`) or not (`false`).	Small, 60.0	60.0
`reason`	String	Reason for the qualification of this product.	Small, 60.0	60.0

Configurator UI Treatment

Output representation of the details of the UI treatments of a product configurator. The details include the product configuration rule actions to override the disable or hide behavior in the UI for product options, product attributes, and attribute picklist values.

JSON Example

```
[
  {
    "details": {
      "attributeId": "0tjxx0000000007AAA",
      "prcId": "0dSxx0000000007EAA",
      "stiId": "0QLxx0000004CU0GAM",
      "attributePicklistValueId": "0v6xx0000000005AAA"
    },
    "uiTreatmentScope": "Bundle",
    "uiTreatmentTarget": "Attribute_Picklist_Value",
    "uiTreatmentType": "Hide"
  },
  {
    "details": {
      "stiId": "ref_f0f2da7b_c431_482d_bf4b_599052f3a2e1"
    },
    "uiTreatmentScope": "Product",
    "uiTreatmentTarget": "Component",
    "uiTreatmentType": "Disable"
```

```
    }
]
```

Property Name	Type	Description	Filter Group and Version	Available Version
`details`	Map<String, Object>	Key-value pair that specifies the items to apply the rules on, which includes these details. • ID of the sales transaction item • ID of the product-related component • ID of the attribute • ID of the attribute picklist value	Small, 62.0	62.0
`uiTreatmentScope`	String	Type of the UI treatment to be performed. Valid values are: • `Product`—UI treatment is applicable to a certain product only. • `Bundle`—UI treatment is applicable to the whole bundle.	Small, 62.0	62.0
`uiTreatmentTarget`	String	Target of the UI treatment. Valid values are: • `Component`—Represents a product option or bundle component. • `Quantity`—Represents a quantity field. • `Attribute`—Represents a certain attribute of the product. • `Attribute_Picklist_Value`—Represents one of the picklist values of a product attribute.	Small, 62.0	62.0
`uiTreatmentType`	String	Type of UI treatment to be performed. Valid values are: • `Hide`—Hide the associated target. • `Disable`—Disable the associated target.	Small, 62.0	62.0

Configurator Unit Of Measure

Output representation of the details of the unit of measure record.

JSON Example

```
{
  "unitOfMeasure": {
    "id": "0hEXR00000000BJ2AY",
    "name": "Litres",
    "roundingMethod": "Down",
    "scale": 2,
    "unitCode": "Ltrs"
  }
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`id`	String	ID of the unit of measure record.	Small, 63.0	63.0
`name`	String	Name of the unit of measure record.	Small, 63.0	63.0
`roundingMethod`	String	Rounding method associated with the unit of measure record.	Small, 63.0	63.0
`scale`	Integer	Scale associated with the unit of measure record.	Small, 63.0	63.0
`unitCode`	String	Unit code associated with the unit of measure record.	Small, 63.0	63.0

Configurator Update Nodes

Output representation of the configuration request details to update nodes.

JSON Example

```
{
  "configuratorMessages": {},
  "configuratorUITreatments": [
    {
      "details": {
        "attributeId": "0tjxx0000000007AAA",
        "prcId": "0dSxx0000000007EAA",
        "stiId": "0QLxx0000004CU0GAM",
        "attributePicklistValueId": "0v6xx0000000005AAA"
      },
      "uiTreatmentScope": "Bundle",
      "uiTreatmentTarget": "Attribute_Picklist_Value",
      "uiTreatmentType": "Hide"
    },
    {
      "details": {
        "stiId": "ref_f0f2da7b_c431_482d_bf4b_599052f3a2e1"
      },
      "uiTreatmentScope": "Product",
      "uiTreatmentTarget": "Component",
```

```
      "uiTreatmentType": "Disable"
    }
  ],
  "errors": [],
  "productQualifications": {
    "01tDU000000EOTCYA4": {
      "isQualified": true
    }
  },
  "success": true
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
configuratorMessages	Map<String, Configurator Message>	Map of the product IDs to the list of configurator messages. Configurator messages are results from any validations, Business Rules Engines (BRE) calls, or Salesforce Pricing calls.	Small, 60.0	60.0
configuratorUITreatments	Configurator UI Treatment[]	Details of the UI treatments that specify the product configuration rule actions to override the disable or hide behavior in the UI for product options, product attributes, and attribute picklist values.	Small, 62.0	62.0
errors	Error Response	List of errors, which contains an error code and a message.	Small, 60.0	60.0
productQualifications	Map<String, Configurator Qualification Context>	Map of the product IDs to the qualification rule execution results.	Small, 60.0	60.0
success	Boolean	Indicates whether the call was successful (`true`) not (`false`).	Small, 60.0	60.0

Configuration Update

Output representation of the details of the updated configuration.

JSON example

This example shows a sample when the update operation is successful.

```
{
  "errors": [],
  "success": true
}
```

This example shows a sample when the update operation has errors.

```
{
  "errors": [
```

```
    {
      "code": "INTERNAL_SERVER_ERROR",
      "message": "INVALID_REFERENCEOBJECTID"
    }
  ],
  "success": false
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
errors	Error Response	List of errors that contains a message and an error code.	Small, 63.0	63.0
success	Boolean	Indicates whether the update operation is successful (true) or not (false)	Small, 63.0	63.0

Error Response

Output representation of the details of the error.

JSON example

```
{
  "errors": [
    {
      "code": "BAD_REQUEST",
      "message": "MISSING_REFERENCEOBJECTID"
    }
  ]
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
code	String	Code of the error.	Small, 63.0	63.0
message	String	Description of the error.	Small, 63.0	63.0

Product Quantity Set Configurator

Output representation of the request details to set product quantity.

JSON Example

```
{
  "configuratorMessages": {},
  "configuratorUITreatments": [
    {
      "details": {
        "attributeId": "0tjxx0000000007AAA",
        "prcId": "0dSxx0000000007EAA",
        "stiId": "0QLxx0000004CU0GAM",
```

```
        "attributePicklistValueId": "0v6xx0000000005AAA"
      },
      "uiTreatmentScope": "Bundle",
      "uiTreatmentTarget": "Attribute_Picklist_Value",
      "uiTreatmentType": "Hide"
    },
    {
      "details": {
        "stiId": "ref_f0f2da7b_c431_482d_bf4b_599052f3a2e1"
      },
      "uiTreatmentScope": "Product",
      "uiTreatmentTarget": "Component",
      "uiTreatmentType": "Disable"
    }
  ],
  "errors": [],
  "productQualifications": {
    "01tDU000000EOTCYA4": {
      "isQualified": true
    }
  },
  "success": true
}
```


Property Name	Type	Description	Filter Group and Version	Available Version
`configuratorMessages`	Map<String, Configurator Message>	Map of the product IDs to the list of configurator messages. Configurator messages are results from any validations, Business Rules Engines (BRE) calls, or Salesforce Pricing calls.	Small, 60.0	60.0
`configuratorUITreatments`	Configurator UI Treatment[]	Details of the UI treatments that specify the product configuration rule actions to override the disable or hide behavior in the UI for product options, product attributes, and attribute picklist values.	Small, 62.0	62.0
`errors`	Error Response	List of errors, which contains an error code and a message.	Small, 60.0	60.0
`productQualifications`	Map<String, Configurator Qualification Context>	Map of the product IDs to the qualification rule execution results.	Small, 60.0	60.0
`success`	Boolean	Indicates whether the call was successful (`true`) not (`false`).	Small, 60.0	60.0

CHAPTER 5 Transaction Management

In this chapter ...

- Quote and Order Capture
- Asset Lifecycle

Automate and optimize the quote creation or order capture experience with the right products and right prices whether its an initial, amendment, or renewal sale.

EDITIONS

Available in: Lightning Experience

Available in: **Enterprise**, **Unlimited**, and **Developer** Editions of Revenue Cloud where Transaction Management is enabled

Quote and Order Capture

Configure, price, and sell products with Quote and Order Capture in Revenue Cloud. Quote and Order Capture supports subscription lifecycles and ensures end-to-end integrity of your quotes and orders.

EDITIONS

Available in: Lightning Experience

Available in: **Enterprise**, **Unlimited**, and **Developer** Editions of Revenue Cloud where Transaction Management is enabled

Quote and Order Capture Standard Objects

The Quote and Order Capture data model provides objects and fields to manage transactions.

Quote and Order Capture Fields on Standard Objects

Quote and Order Capture adds standard and custom fields to some standard Salesforce objects. These fields are available only in orgs where Quote and Order Capture is enabled.

Quote and Order Capture Tooling API Objects

Tooling API exposes metadata used in developer tooling that you can access through REST or SOAP. Tooling API's SOQL capabilities for many metadata types allow you to retrieve smaller pieces of metadata.

Quote and Order Capture Platform Event

Use the QuoteSaveEvent event to notify subscribers after saving of a quote is processed.

Quote and Order Capture Business APIs

Use the Quote and Order Capture Business APIs to fetch instant pricing data on a quote or an order, to create a quote, or to create an order.

Quote and Order Capture Apex Reference

Use built-in Apex classes and interfaces grouped by namespace.

Quote and Order Capture Standard Invocable Actions

Learn more about the standard invocable actions available with Quote and Order Capture.

Quote and Order Capture Metadata API Types

Metadata API enables you to access some types and feature settings that you can customize in the user interface.

SEE ALSO:

Salesforce Help: Permissions for Quote and Order Capture

Quote and Order Capture Standard Objects

The Quote and Order Capture data model provides objects and fields to manage transactions.

ApprovalSubmission

Represents the instance of an approval request that's submitted for a record of the related object. This object is available in API version 62.0 and later.

ContractItemPrice

Represents an object that's used to capture a price for a product on a contract. This object is available in API version 61.0 and later.

ContractItemPriceAdjTier

Represents the tiers of a price adjustment to a product on a contract. This object is available in API version 63.0 and later.

ContractItemPriceHistory

Represents the history of changes to the values in the fields of a ContractItemPrice object. This object is available in API version 61.0 and later.

OrderDeliveryMethod

Shows the customizations and options that a buyer selected for their delivery method. This object is available in API version 48.0 and later.

OrderItemAttribute

Represents a virtual object which is associated with the Attribute Value Store object. This object is available in API version 60.0 and later.

OrderItemDetail

Represents information about the quantity and price details of an order product. This object is available in API version 60.0 and later.

OrderItemRateAdjustment

Represents the negotiated rate adjustment for an order product. This object is available in API version 62.0 and later.

OrderItemRateCardEntry

Represents the catalog and negotiated rates of a usage metric associated with an order item that's used to charge overage consumption. This object is available in API version 62.0 and later.

SalesTransactionType

Represents the type of the sales transaction. This object is available in API version 61.0 and later.

QuoteAction

Indicates the type of sales transaction that's being quoted; for example, a renewal sale. This object is available in API version 59.0 and later.

QuoteLineDetail

Represents information about the quantity and price detail of a quote line item. This object is available in API version 60.0 and later.

QuoteLineRateAdjustment

Represents the negotiated rate adjustment for a quote line item. This object is available in API version 62.0 and later.

QuoteLineRateCardEntry

Represents the catalog and negotiated rates of a usage resource associated with a quote line item that's used to charge overage consumption. This object is available in API version 62.0 and later.

SEE ALSO:

Object Reference for the Salesforce Platform: Overview of Salesforce Objects and Fields

SOAP API Developer Guide: Introduction to SOAP API

ApprovalSubmission

Represents the instance of an approval request that's submitted for a record of the related object. This object is available in API version 62.0 and later.

Supported Calls

`describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

This object is available in Enterprise, Unlimited, and Developer Editions of Revenue Cloud.

Fields

Field	Details
`Comments`	**Type** textarea **Properties** Nillable **Description** The comments about the request that's submitted for approval.
`DoesSendApprovalEmail`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Required. Indicates whether approval request emails are sent to approvers and delegates (`true`) or not (`false`). The default value is `false`.
`FlowOrchestrationInstanceId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of the flow orchestration instance record that's associated with the approval. This field is a relationship field. **Relationship Name** FlowOrchestrationInstance **Refers To** FlowOrchestrationInstance
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The auto-generated name for the approval submission.

Field	Details
OwnerId	**Type** reference **Properties** Filter, Group, Sort **Description** The ID of the user or the group that owns the approval submission record. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
RelatedRecordId	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** Required. The ID of the related record that's submitted for approval. This field is a polymorphic relationship field. **Relationship Name** RelatedRecord **Refers To** Account, AdAvailabilityViewConfig, Address, AnalyticsUserAttrFuncTkn, ApprovalSubmission, ApprovalSubmissionDetail, ApprovalWorkItem, Asset, AssetAction, AssetActionSource, AssetContractRelationship, AssetRateAdjustment, AssetRateCardEntry, AssetRelationship, AssetStatePeriod, AssociatedLocation, AsyncOperationTracker, AttrPicklistExcludedValue, AttributeAdjustmentCondition, AttributeBasedAdjRule, AttributeBasedAdjustment, AttributeCategory, AttributeCategoryAttribute, AttributeDefinition, AttributePicklist, AttributePicklistValue, AuthorizationForm, AuthorizationFormConsent, AuthorizationFormDataUse, AuthorizationFormText, BatchJob, BatchJobPart, BatchJobPartFailedRecord, BundleBasedAdjustment, BusinessBrand, Case, CaseComment, ChannelProgram, ChannelProgramLevel, ChannelProgramMember, CollaborationGroup, CommSubscription, CommSubscriptionChannelType, CommSubscriptionConsent, CommSubscriptionTiming, Contact, ContactPointAddress, ContactPointConsent, ContactPointEmail, ContactPointPhone, ContactPointTypeConsent, ContactRequest, ContextDefinitionSync, Contract, ContractItemPrice, ContractItemPriceAdjTier, CostBook, CostBookEntry, Customer, DTRecordsetReplica, DataUseLegalBasis, DataUsePurpose, DecisionTblFileImportData, DelegatedAccount, DocGenerationQueryResult, DocTemplateSectionCondition, DocumentEnvelope, DocumentGenerationProcess, DocumentRecipient, DocumentTemplate, DocumentTemplateContentDoc, DocumentTemplateSection, DocumentTemplateToken, DuplicateRecordItem, DuplicateRecordSet, EmailMessage, EngagementChannelType, ExpressionSetConstraintObj, ExternalEventMapping, FlowOrchestrationInstance, FulfillmentOrder,

Field	Details
	FulfillmentOrderItemAdjustment, FulfillmentOrderItemTax, FulfillmentOrderLineItem, GeneratedDocument, GeneratedDocumentSection, Idea, Image, Individual, IntegrationProviderDcsnRqmt, IntegrationProviderExecution, Lead, Location, LocationTrustMeasure, ManagedContentVariant, ObjectStateDefinition, ObjectStateTransition, ObjectStateValue, Obligation, Opportunity, OpportunityRelatedDeleteLog, Order, OrderAction, OrderAdjustmentGroup, OrderDeliveryGroup, OrderDeliveryMethod, OrderItem, OrderItemAdjustmentLineItem, OrderItemDetail, OrderItemRateAdjustment, OrderItemRateCardEntry, OrderItemRecipient, OrderItemRelationship, OrderItemTaxLineItem, OrgMetricScanResult, OrgMetricScanSummary, Organization, PartnerFundAllocation, PartnerFundClaim, PartnerFundRequest, PartnerMarketingBudget, PartyConsent, PriceBookEntryDerivedPrice, PriceBookRateCard, PricingAdjBatchJob, PricingAdjBatchJobLog, PricingApiExecution, PricingProcessExecution, ProcessException, Product2, ProductAttributeDefinition, ProductCatalog, ProductCategory, ProductCategoryDisqual, ProductCategoryProduct, ProductCategoryQualification, ProductClassification, ProductClassificationAttr, ProductComponentGroup, ProductComponentGrpOverride, ProductConfigFlowAssignment, ProductConfigurationFlow, ProductConfigurationRule, ProductDisqualification, ProductPriceHistoryLog, ProductPriceRange, ProductQualification, ProductRampSegment, ProductRelComponentOverride, ProductUsageGrant, ProfileSkill, ProfileSkillEndorsement, ProfileSkillUser, PromptAction, PromptError, QuickText, QuickTextUsage, Quote, QuoteLineDetail, QuoteLineItem, QuoteLineItemRecipient, QuoteLineRateAdjustment, QuoteLineRateCardEntry, RateAdjustmentByAttribute, RateAdjustmentByTier, RateCard, RateCardEntry, RatingFrequencyPolicy, SalesTransactionType, Seller, Shipment, ShipmentItem, Site, SocialPersona, SocialPost, Solution, StreamingChannel, TableauHostMapping, Topic, UnitOfMeasure, UnitOfMeasureClass, UsageGrantRenewalPolicy, UsageGrantRolloverPolicy, UsageResource, UsageResourceBillingPolicy, User, UserEsignVendorIdentifier, UserLicense, UserLocalWebServerIdentity, UserProvisioningRequest, WorkBadge, WorkBadgeDefinition, WorkOrder, WorkOrderLineItem, WorkThanks
`RelatedRecordObjectName`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** Required. The type of record that was submitted for approval.
`Status`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Required. The status of the approval. Valid values are: • `Approved` • `Canceled`

Field	Details
	• `Errored` • `InProgress` • `Recalled` • `Rejected` • `Suspended`
`SubmittedById`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** Required. The ID of the user who submitted the record for approval. This field is a relationship field. **Relationship Name** SubmittedBy **Refers To** User

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ApprovalSubmissionOwnerSharingRule on page 1682
Sharing rules are available for the object.

ApprovalSubmissionShare on page 1684
Sharing is available for the object.

ContractItemPrice

Represents an object that's used to capture a price for a product on a contract. This object is available in API version 61.0 and later.

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.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

This object is available in Enterprise, Unlimited, and Developer Editions of Revenue Cloud.

Fields

Field	Details
`AdjustmentMethod`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Method used to apply discount. Valid values are: • `Range`—Apply the discount to all items after you reach the discount tier. For example, suppose that you give a 10% discount for 50 or more items. If a customer orders 50 products, and the type is `range`, apply the 10% discount to all 50 items. • `Slab`—Apply discounts in tiers. For example, suppose that you order 30 products, and the type is `slab`, you can apply a 10% discount to units 1-9, a 20% discount to units 10-19, and a 30% discount to units 20-30.
`ContractId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** ID of the contract. This field is a relationship field. **Relationship Name** Contract **Relationship Type** Master-detail **Refers To** Contract (the master object)
`DiscountType`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Type of discount to apply, which is a percentage of the price or an amount. Valid values are: • `AdjustmentAmount` • `AdjustmentPercentage`

Field	Details
DiscountValue	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** Value of the discount to apply based on the discount type.
EndDate	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** End date and time of the relationship between the contract and contract item price.
ItemId	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** ID of the product or product category related to a price in a contract. This field is a polymorphic relationship field. **Relationship Name** Item **Relationship Type** Lookup **Refers To** Product2, ProductCategory
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Timestamp when the current user last accessed this record, a record related to this record, or a list view.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort

Field	Details
	Description Timestamp when the current user last viewed this record or list view. If this value is null, the user accessed this record or list view (`LastReferencedDate`) but didn't view it.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Auto-generated number assigned to the contract item price. (Read Only)
`Price`	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** Unit price for the product sold as part of the contract.
`ProductSellingModelId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Product selling model for the product associated with the price. This field is a relationship field. **Relationship Name** ProductSellingModel **Relationship Type** Lookup **Refers To** ProductSellingModel
`SellingModelType`	**Type** picklist **Properties** Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort **Description** Selling mode type to specify whether the product is sold as a one-time sale, an evergreen subscription, or a subscription with a defined term. Valid values are: • `Evergreen`

Field	Details
	• `OneTime` • `TermDefined` This value is derived from the product selling model.
`StartDate`	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** Start date and time of the relationship between the contract and contract item price.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ContractItemPriceHistory on page 635
History is available for tracked fields of the object.

ContractItemPriceAdjTier

Represents the tiers of a price adjustment to a product on a contract. This object is available in API version 63.0 and later.

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.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

This object is available with Revenue Cloud.

Fields

Field	Details
`ContractItemPriceId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The contract item price ID associated with the contract item price adjustment tier.

Field	Details
	This field is a relationship field. **Relationship Name** ContractItemPrice **Relationship Type** Master-detail **Refers To** ContractItemPrice (the master object)
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed this record, a record related to this record, or a list view.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Timestamp when the current user last viewed this record or list view. If this value is null, the user accessed this record or list view (`LastReferencedDate`) but didn't view it.
`LowerBound`	**Type** double **Properties** Create, Filter, Sort, Update **Description** The minimum quantity for the adjustment to be applicable.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name of the record.
`TierType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update

Field	Details
	Description The type of adjustment tier. Valid values are: • `AdjustmentAmount` • `AdjustmentPercentage` • `OverrideAmount`
`TierValue`	**Type** double **Properties** Create, Filter, Sort, Update **Description** The price adjustment value.
`UpperBound`	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The maximum quantity for the adjustment to be applicable.

Associated Objects

This object has associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ContractItemPriceAdjTierFeed
Feed tracking is available for the object.

ContractItemPriceAdjTierHistory
History is available for tracked fields of the object.

ContractItemPriceHistory

Represents the history of changes to the values in the fields of a ContractItemPrice object. This object is available in API version 61.0 and later.

Supported Calls

`describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

This object is available in Enterprise, Unlimited, and Developer Editions of Revenue Cloud.

Fields

Field	Details
ContractItemPriceId	**Type** reference **Properties** Filter, Group, Sort **Description** ID of the ContractItemPrice record. This field is a relationship field. **Relationship Name** ContractItemPrice **Relationship Type** Lookup **Refers To** ContractItemPrice
DataType	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** Data type of the field that was changed. Valid values are: • Address • AnyType • AutoNumber • Base64 • BitVector • Boolean • Content • Currency • DataCategoryGroupReference • DateOnly • DateTime • Division • Double • DynamicEnum • Email • EncryptedBase64

Field	Details
	• EncryptedText • EntityId • EnumOrId • ExternalId • Fax • File • HtmlMultiLineText • HtmlStringPlusClob • InetAddress • Json • Location • MultiEnum • MultiLineText • Namespace • Percent • PersonName • Phone • Raw • RecordType • SfdcEncryptedText • SimpleNamespace • StringPlusClob • Switchable_PersonName • Text • TimeOnly • Url • YearQuarter
Field	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Name of the field that was changed. Possible values are: • AdjustmentMethod • Contract • DiscountType

Field	Details
	• `DiscountValue` • `EndDate` • `Item` • `Name` • `Price` • `ProductSellingModel` • `StartDate` • `created` • `customPersonMerged` • `feedEvent` • `individualMerged` • `locked`—Record locked. • `ownerAccepted`—Owner (Accepted) • `ownerAssignment`—Owner (Assignment) • `unlocked`—Record unlocked.
`NewValue`	**Type** anyType **Properties** Nillable, Sort **Description** New value of the field that was changed.
`OldValue`	**Type** anyType **Properties** Nillable, Sort **Description** Latest value of the field before it was changed.

OrderDeliveryMethod

Shows the customizations and options that a buyer selected for their delivery method. This object is available in API version 48.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

To access Commerce Orders entities, your org must have a Salesforce Order Management license. Commerce Orders entities are available only in Lightning Experience.

Fields

Field	Details
`Carrier`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The carrier that the buyer chose for their delivery method. Developers must add values to this field.
`ClassOfService`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The carrier class of service that the buyer chose for their delivery method. Developers must add values to this field.
`Description`	**Type** textarea **Properties** Create, Nillable, Update **Description** Description of the delivery method.
`IsActive`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Assign new delivery groups to active delivery methods.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed a record related to this record.

Field	Details
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (LastReferencedDate) and not viewed.
Name	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** Required. Default name of this record.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The user who owns an order delivery method record.
ProductId	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Optional. This product represents a delivery charge order product for a delivery using this delivery method. For example, you could create a product that represents an overnight express charge and assign it to an overnight express delivery method.
ReferenceNumber	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Reference number for an external delivery method.
ShippingCarrierMethod	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description Optional. A specific shipping service provided by a shipping carrier, such as Ground, 2Day, and NextDay. Depends on the range of transit times available for each carrier.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

OrderDeliveryMethodChangeEvent (API version 62.0)
Change events are available for the object.

OrderItemAttribute

Represents a virtual object which is associated with the Attribute Value Store object. This object is available in API version 60.0 and later.

Supported Calls

`create(),delete(),describeLayout(),describeSObjects(),query(),retrieve(),update(),upsert()`

Special Access Rules

This object is available in Enterprise, Unlimited, and Developer Editions of Revenue Cloud.

Fields

Field	Details
`AttributeDefinitionId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The ID of the attribute definition for this order item attribute. This field is a relationship field. **Relationship Name** AttributeDefinition **Relationship Type** Lookup **Refers To** AttributeDefinition
`AttributeName`	**Type** string

Field	Details
	Properties Filter, Group, idLookup, Nillable, Sort **Description** The name given to order item attribute.
`AttributePicklistValueId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ID of the attribute picklist value if the attribute is a picklist type. This field is a relationship field. **Relationship Name** AttributePicklistValue **Relationship Type** Lookup **Refers To** AttributePicklistValue
`AttributeValue`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Stores the value of the order item attribute. For example 5-TB storage.
`ExternalId`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** An auto-generated ID of the attribute record saved in an external system (for example an HBase database).
`OrderItemId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The parent order item associated with the order item attribute. This field is a relationship field.

Field	Details
	Relationship Name OrderItem **Relationship Type** Lookup **Refers To** OrderItem

OrderItemDetail

Represents information about the quantity and price details of an order product. This object is available in API version 60.0 and later.

Supported Calls

`describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

This object is available in Enterprise, Unlimited, and Developer Editions of Revenue Cloud.

Fields

Field	Details
BillingReference	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** Reference to the original order item for which this amend or cancel record is created.
`LineNumber`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The line number of the detail record.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort

Field	Details
	Description The name of the record.
`NetUnitPrice`	**Type** currency **Properties** Filter, Sort **Description** The unit price of the item that's calculated after applying any discounts and before tax calculation.
`OrderItemId`	**Type** reference **Properties** Filter, Group, Sort **Description** The ID of the related order item. This field is a relationship field. **Relationship Name** OrderItem **Relationship Type** Lookup **Refers To** OrderItem
`PriceWaterfallIdentifier`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The price waterfall identifier generated by Salesforce Pricing that's associated with the pricing of the detail record.
`Quantity`	**Type** double **Properties** Filter, Sort **Description** The quantity specified for the item in the detail record.

Field	Details
ReferenceDate	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The reference date of the item. For example, the start date of the subscription that's associated with the detail record.
ReferenceNumber	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The reference number of the item. For example, the order number that's associated with the detail record.
TotalLineAmount	**Type** currency **Properties** Filter, Sort **Description** The net total price of the order product, before price adjustments, inclusive of quantity and subscription term.
TotalPrice	**Type** currency **Properties** Filter, Sort **Description** The total price of the item in the detail record that's calculated using the quantity, net unit price, and after applying the pricing terms.
UnitPrice	**Type** currency **Properties** Filter, Sort **Description** The unit price of the item before any discounts or tax calculation.

OrderItemRateAdjustment

Represents the negotiated rate adjustment for an order product. This object is available in API version 62.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

This object is available with Revenue Cloud.

Fields

Field	Details
`AdjustmentType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Specifies the type of rate adjustment. Possible values are: • `Amount` • `Override` • `Percentage`
`AdjustmentValue`	**Type** double **Properties** Create, Filter, Sort, Update **Description** The value of the adjustment.
`LowerBound`	**Type** double **Properties** Create, Filter, Sort, Update **Description** The minimum quantity for the adjustment to be applicable.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** The name of the order item rate adjustment.

Field	Details
`OrderItemRateCardEntryId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The parent order item rate card entry associated with the order item rate adjustment. This field is a relationship field. **Relationship Name** OrderItemRateCardEntry **Relationship Type** Master-detail **Refers To** OrderItemRateCardEntry (the master object)
`UpperBound`	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The quantity below which the adjustment must be applicable. For example, if you want the adjustment to be applicable when the quantity is 99 or less, set this value to 100.

OrderItemRateCardEntry

Represents the catalog and negotiated rates of a usage metric associated with an order item that's used to charge overage consumption. This object is available in API version 62.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

This object is available with Revenue Cloud.

Fields

Field	Details
`Name`	**Type** string

Field	Details
	Properties Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** An auto-generated number assigned to the order item rate card entry record.
`NegotiatedRate`	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The base negotiated rate used to charge overage consumption.
`OrderItemId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The parent order item associated with the order item rate card entry. This field is a relationship field. **Relationship Name** OrderItem **Relationship Type** Master-detail **Refers To** OrderItem (the master object)
`RateCardEntryId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The rate card entry containing catalog rates that's associated with the order item rate card entry. This field is a relationship field. **Relationship Name** RateCardEntry **Refers To** RateCardEntry
`RateCardId`	**Type** reference

Field	Details
	Properties Filter, Group, Nillable, Sort **Description** The rate card associated with the order item rate card entry. This field is a relationship field. **Relationship Name** RateCard **Refers To** RateCard
`RateUnitOfMeasureId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The standard unit of measure containing the unit for the negotiated rate that's associated with the order item rate card entry. This field is a relationship field. **Relationship Name** RateUnitOfMeasure **Refers To** UnitOfMeasure
`UsageResourceId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The usage resource associated with the order item rate card entry. This field is a relationship field. **Relationship Name** UsageResource **Refers To** UsageResource

SalesTransactionType

Represents the type of the sales transaction. This object is available in API version 61.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

This object is available in Enterprise, Unlimited, and Developer Editions of Revenue Cloud.

Fields

Field	Details
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed this record, a record related to this record, or a list view.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed this record or list view. If this value is null, the user might have only accessed this record or list view but not viewed it directly.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** The name of the sales transaction type.

QuoteAction

Indicates the type of sales transaction that's being quoted; for example, a renewal sale. This object is available in API version 59.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

This object is available in orgs with Subscription Management or Revenue Cloud. It's also available in Industries Automotive and Industries Field Service.

Fields

Field	Details
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed this record indirectly, for example, through a list view or related record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed this record or list view. If this value is null, the user accessed this record or list view indirectly, but didn't view it.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name given to the quote action.
`QuoteId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The quote related to this quote action. This field is a relationship field. **Relationship Name** Quote **Relationship Type** Lookup

Field	Details
	Refers To Quote
`SourceAssetId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The asset changed by this sales transaction. For example, if the quote action is a quantity amendment, this field contains the ID of the asset that's amended. This field is a relationship field. **Relationship Name** SourceAsset **Relationship Type** Lookup **Refers To** Asset
`Type`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort **Description** The type of sales transaction that the related quote is for. Valid values are: • `Add` • `Amend` • `Cancel` • `No Change` • `Renew`

QuoteLineDetail

Represents information about the quantity and price detail of a quote line item. This object is available in API version 60.0 and later.

Supported Calls

`describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

This object is available in Enterprise, Unlimited, and Developer Editions of Revenue Cloud.

Fields

Field	Details
`BillingReference`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** Reference to the original order item for which this amend or cancel record is created.
`LineNumber`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The line number of the detail record.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name of the record.
`NetUnitPrice`	**Type** currency **Properties** Filter, Sort **Description** The unit price of the item that's calculated after applying any discounts and before tax calculation.
`PriceWaterfallIdentifier`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The price waterfall identifier generated by Salesforce Pricing that's associated with the pricing of the detail record.

Field	Details
`Quantity`	**Type** double **Properties** Filter, Sort **Description** The quantity specified for the item in the detail record.
`QuoteLineItemId`	**Type** reference **Properties** Filter, Group, Sort **Description** The ID of the related quote line. This field is a relationship field. **Relationship Name** QuoteLineItem **Relationship Type** Lookup **Refers To** QuoteLineItem
`ReferenceDate`	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The reference date of the item. For example, the start date of the subscription that's associated with the detail record.
`ReferenceNumber`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The reference number of the item. For example, the order number that's associated with the detail record.
`TotalLineAmount`	**Type** currency **Properties** Filter, Sort

Field	Details
	Description The net total price of the order product, before price adjustments, inclusive of quantity and the subscription term.
`TotalPrice`	**Type** currency **Properties** Filter, Sort **Description** The total price of the item in the detail record that's calculated using the quantity, net unit price, and after applying the pricing terms.
`UnitPrice`	**Type** currency **Properties** Filter, Sort **Description** The unit price of the item before any discounts or tax calculation.

QuoteLineRateAdjustment

Represents the negotiated rate adjustment for a quote line item. This object is available in API version 62.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

This object is available with Revenue Cloud.

Fields

Field	Details
`AdjustmentType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Specifies the type of rate adjustment.

Field	Details
	Possible values are: • `Amount` • `Override` • `Percentage`
`AdjustmentValue`	**Type** double **Properties** Create, Filter, Sort, Update **Description** The value of the adjustment.
`LowerBound`	**Type** double **Properties** Create, Filter, Sort, Update **Description** The minimum quantity for the adjustment to be applicable.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** The name of the quote line rate adjustment record.
`QuoteLineRateCardEntryId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The parent quote line rate card entry associated with the quote line rate adjustment. This field is a relationship field. **Relationship Name** QuoteLineRateCardEntry **Relationship Type** Master-detail **Refers To** QuoteLineRateCardEntry (the master object)

Field	Details
`UpperBound`	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The quantity below which the adjustment must be applicable. For example, if you want the adjustment to be applicable when the quantity is 99 or less, set this value to 100.

QuoteLineRateCardEntry

Represents the catalog and negotiated rates of a usage resource associated with a quote line item that's used to charge overage consumption. This object is available in API version 62.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

This object is available with Revenue Cloud.

Fields

Field	Details
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** An auto-generated number assigned to the quote line rate card entry record.
`NegotiatedRate`	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The base negotiated rate used to charge overage consumption.
`QuoteLineItemId`	**Type** reference

Field	Details
	Properties Create, Filter, Group, Sort **Description** The parent quote line item associated with the quote line rate card entry. This field is a relationship field. **Relationship Name** QuoteLineItem **Relationship Type** Master-detail **Refers To** QuoteLineItem (the master object)
`RateCardEntryId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The rate card entry containing catalog rates that's associated with the quote line rate card entry. This field is a relationship field. **Relationship Name** RateCardEntry **Refers To** RateCardEntry
`RateCardId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The rate card associated with the quote line rate card entry. This field is a relationship field. **Relationship Name** RateCard **Refers To** RateCard
`RateUnitOfMeasureId`	**Type** reference **Properties** Filter, Group, Nillable, Sort

Field	Details
	Description The standard unit of measure containing the unit for the negotiated rate that's associated with the quote line rate card entry. This field is a relationship field. **Relationship Name** RateUnitOfMeasure **Refers To** UnitOfMeasure
`UsageResourceId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The usage resource associated with the quote line rate card entry. This field is a relationship field. **Relationship Name** UsageResource **Refers To** UsageResource

Quote and Order Capture Fields on Standard Objects

Quote and Order Capture adds standard and custom fields to some standard Salesforce objects. These fields are available only in orgs where Quote and Order Capture is enabled.

Quote and Order Capture Fields on Object State Definition

Standard and custom fields extend the standard Object State Definition object for use in Quote and Order Capture to represent the object state model for a particular status field for an entity. This object is available in API version 60.0 and later.

Quote and Order Capture Fields on Object State Transition

Standard and custom fields extend the standard Object State Transition object for use in Quote and Order Capture to define the valid transition between two statuses. This object is available in API version 60.0 and later.

Quote and Order Capture Fields on Object State Value

Standard and custom fields extend the standard Object State Transition object for use in Quote and Order Capture This object is available in API version 60.0 and later.

Quote and Order Capture Fields on Order

Standard and custom fields extend the standard Order object for use in Quote and Order Capture.

Quote and Order Capture Fields on OrderItem

Standard and custom fields extend the standard Order Item object for use in Quote and Order Capture.

Quote and Order Capture Fields on Order Action

Standard and custom fields extend the standard Order Action object for use in Quote and Order Capture. This object is available in API version 55.0 and later.

Quote and Order Capture fields on Order Item Relationship

Standard and custom fields extend the standard Order Item Relationship object for use in Quote and Order Capture. This object is available in API version 58.0 and later.

Quote and Order Capture Fields on Quote

Standard and custom fields extend the standard Quote object for use in Quote and Order Capture to represent information about quotes. This object is available in API version 60.0 and later.

Quote and Order Capture Fields on Quote Line Item

Standard and custom fields extend the standard Quote Line Item object for use in Quote and Order Capture to represent information about line items in a quote. This object is available in API version 60.0 and later.

Quote and Order Capture Fields on Quote Document

Standard and custom fields extend the standard Quote Document object for use in Quote and Order Capture to represent information about quote documents. This object is available in API version 61.0 and later.

Quote and Order Capture Fields on Object State Definition

Standard and custom fields extend the standard Object State Definition object for use in Quote and Order Capture to represent the object state model for a particular status field for an entity. This object is available in API version 60.0 and later.

Fields

Field	Details
`AppUsageType`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** This field indicates under which AppUsageType the transition applies to. For example, ObjectStateDefinition associated with "Revenue Lifecycle Management" AppUsageType will apply to quotes, assets, or orders associated with "Revenue Lifecycle Management".

Quote and Order Capture Fields on Object State Transition

Standard and custom fields extend the standard Object State Transition object for use in Quote and Order Capture to define the valid transition between two statuses. This object is available in API version 60.0 and later.

Fields

Field	Details
`CustomPermissionId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ID of the associated custom permission. This field is a relationship field. **Relationship Name** CustomPermission **Relationship Type** Lookup **Refers To** CustomPermission

Quote and Order Capture Fields on Object State Value

Standard and custom fields extend the standard Object State Transition object for use in Quote and Order Capture This object is available in API version 60.0 and later.

Fields

Field	Details
`CustomPermissionId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ID of the associated custom permission. This field is a relationship field. **Relationship Name** CustomPermission **Relationship Type** Lookup **Refers To** CustomPermission

Quote and Order Capture Fields on Order

Standard and custom fields extend the standard Order object for use in Quote and Order Capture.

Fields

Field	Details
`CalculationStatus`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** The status of the price and tax calculations for the order. Valid values are: • `CompletedWithPricing`—Indicates that pricing is complete and tax will now be calculated. • `CompletedWithTax`—Indicates that pricing and tax calculation are complete. • `CompletedWithoutPricing`—Indicates that pricing and tax calculation were skipped. For a sales rep, this value appears as `Unknown` on the order page. • `ConfigurationFailed`—Indicates that configuration failed. Available in API version 62.0 • `ConfigurationInProgress`—Indicates that the configuration is in progress. Available in API version 62.0 • `OrderRequestFailed`—Indicates that the requested order changes weren't saved. Available in API version 62.0 • `OrderRequestPartiallySaved`—Indicates that the requested order changes were partially saved. Available in API version 62.0 • `PriceCalculationFailed`—Indicates that pricing failed. • `ReconciliationFailed`—Indicates that the arrangement of order data failed. Available in API version 62.0 • `ReconciliationInProgress`—Indicates that the arrangement of data is in progress. For a sales rep, this value appears as `Saving` on the order page. Available in API version 62.0 • `SaveFailedOrIncomplete`—Indicates that the recent changes to the order weren't saved. For a sales rep, this value appears as `Some Records Weren't Saved` on the order page. • `TaxCalculationFailed`—Indicates that pricing is complete but tax calculation failed. • `TaxCalculationWaiting`—Indicates that pricing is complete and a request is sent to the tax engine, but tax calculation isn't complete. This read-only field is available in API version 61.0 and later.

Field	Details
`DiscountPercent`	**Type** percent **Properties** Filter, Nillable, Sort **Description** The percentage of discount applied to the order. Available in API version 60.0 and later.
`FulfillmentPlanId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The unique ID of the fulfillment plan associated with the order. This field is a relationship field. This field is available only in orgs where Dynamic Revenue Orchestrator is enabled. Available in API version 60.0 and later. **Relationship Name** FulfillmentPlan **Relationship Type** Lookup **Refers To** FulfillmentPlan
`LastPricedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date when the order price was last calculated. Available in API version 60.0 and later.
`OrchestrationSbmsStatus`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** The status of order submission for orchestration. This field is available only in orgs where Dynamic Revenue Orchestrator is enabled. Valid values are: • `Completed` • `Rejected`

Field	Details
	• `Submitted` This read-only field is available in API version 61.0 and later.
`OriginalActionType`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** For internal use only. Available in API version 61.0 and later.
`SalesTransactionTypeId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The foreign key to the Sales Transaction Type object. Available in API version 61.0 and later. This field is a relationship field. **Relationship Name** SalesTransactionType **Relationship Type** Lookup **Refers To** SalesTransactionType
`TotalRoundedLineAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total amount of all line items in an order without pricing adjustments, such as discounts or tax calculations. Available in API version 60.0 and later.
`ValidationResult`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Specifies whether the order was configured and priced. Orders can be activated only after they're configured and priced. Valid value is:

Field	Details
	• `TransactionIncomplete`—Indicates that the order wasn't configured and priced. Available in API version 61.0 and later.

Quote and Order Capture Fields on OrderItem

Standard and custom fields extend the standard Order Item object for use in Quote and Order Capture.

Fields

Field	Details
`CustomProductName`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The custom name of a product that's used to override the product name. Available in API version 61.0 and later.
`Discount`	**Type** percent **Properties** Create, Filter, Nillable, Sort, Update **Description** The manual discount percentage for the order item. Available in API version 60.0 and later.
`DiscountAmount`	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The manual discount amount for the order item. Available in API version 60.0 and later.
`EffectiveGrantDate`	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The date on which the resources associated with the order item are granted.

Field	Details
EndQuantity	**Type** double **Properties** Filter, Nillable, Sort **Description** The revised quantity of the item after adjusting changes. The field is read-only. It's calculated by adding the Start Quantity and the Quantity fields. Available in API version 60.0 and later.
NetTotalPrice	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The total price after applying all price adjustments. Available in API version 60.0 and later.
OrderItemGroupId	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The order item group associated with the order item. This field is a relationship field. **Relationship Name** OrderItemGroup **Refers To** OrderItemGroup
PartnerDiscountPercent	**Type** percent **Properties** Create, Filter, Nillable, Sort, Update **Description** The discount percentage given to the partner for the order item. Available in API version 60.0 and later.
PartnerUnitPrice	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update

Field	Details
	Description The unit price after applying the discount given to the partner for the order item. Available in API version 60.0 and later.
`PriceWaterfallIdentifier`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The price waterfall identifier generated by Salesforce Pricing that's associated with the pricing of this order item record. Available in API version 60.0 and later.
`StartQuantity`	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The quantity available on the order item start date. Available in API version 60.0 and later.
`Status`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The status of the order item. Available in API version 60.0 and later.
`SubscriptionTerm`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The number of terms in the subscription for the item. The unit of the subscription term is stored in the PricingTermUnit field of this order item's related product selling model record (OrderItem.ProductSellingModel.PricingTermUnit). Available in API version 61.0 and later.
`ValidationResult`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Specifies whether the order item was configured and priced.

Field	Details
	An order can be activated only after all its order items are configured and priced. Valid value is: • `Warning`—Indicates that the order item isn't configured and priced. Available in API version 60.0 and later.

Quote and Order Capture Fields on Order Action

Standard and custom fields extend the standard Order Action object for use in Quote and Order Capture. This object is available in API version 55.0 and later.

Fields

Field	Details
`Type`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The business action that created the order product. Valid values are: • `Add` • `Amend` • `Cancel` • `No Change`—A child product was added to the bundle, but the top-level product in the bundle was otherwise unchanged. • `Renew`

Quote and Order Capture fields on Order Item Relationship

Standard and custom fields extend the standard Order Item Relationship object for use in Quote and Order Capture. This object is available in API version 58.0 and later.

Fields

Field	Details
`ProductRelatedComponentId`	**Type** reference

Field	Details
	Properties Create, Filter, Group, Nillable, Sort **Description** The ID of the product that is included in a product bundle, a set, or a product and an add-on. This field is a relationship field. **Relationship Name** ProductRelatedComponent **Relationship Type** Lookup **Refers To** ProductRelatedComponent

Quote and Order Capture Fields on Quote

Standard and custom fields extend the standard Quote object for use in Quote and Order Capture to represent information about quotes. This object is available in API version 60.0 and later.

Fields

Field	Details
`LastPricedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date when the quote is last priced.
`PartnerAccountId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** ID of the related partner account. This field is a relationship field. **Relationship Name** PartnerAccount **Relationship Type** Lookup **Refers To** Account

Field	Details
`ValidationResult`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Specifies whether the quote was configured and priced. Quotes can be activated only after they're configured and priced. Valid values are: • `TransactionIncomplete`—Specifies that the quote wasn't configured and priced. Available in API version 61.0 and later.

Quote and Order Capture Fields on Quote Line Item

Standard and custom fields extend the standard Quote Line Item object for use in Quote and Order Capture to represent information about line items in a quote. This object is available in API version 60.0 and later.

Fields

Field	Details
`DiscountAmount`	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** Specifies the fixed amount discount to apply to the quote line item.
`EffectiveGrantDate`	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The date on which the resources associated with the quote line item are granted.
`EndQuantity`	**Type** double **Properties** Filter, Nillable, Sort **Description** The quantity available on the quote line item end date. The field is read-only. It is calculated by adding the Start Quantity and the existing Quantity fields.

Field	Details
PartnerDiscountPercent	**Type** percent **Properties** Create, Filter, Nillable, Sort, Update **Description** The discount percentage given to the partner for the quote line.
PartnerUnitPrice	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The unit price after discount given to the partner for the quote line.
PriceWaterfallIdentifier	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The price waterfall identifier generated by Salesforce Pricing that's associated with the pricing of the detail record.
StartQuantity	**Type** double **Properties** Create, Filter, Nillable, Sort **Description** The quantity available on the quote line item start date.
TaxTreatmentId	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of the tax treatment associated with the quote line item. This field is a relationship field. **Relationship Name** TaxTreatment **Relationship Type** Lookup

Field	Details
	Refers To TaxTreatment
`ValidationResult`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Specifies whether the quote line item is configured and priced. A quote can be activated only after all its quote line items are configured and priced. Valid value is: • `Warning`—Indicates that the quote line item isn't configured and priced. Available in API version 60.0 and later.

Quote and Order Capture Fields on Quote Document

Standard and custom fields extend the standard Quote Document object for use in Quote and Order Capture to represent information about quote documents. This object is available in API version 61.0 and later.

Fields

Field	Details
`Document Template`	**Type** String **Properties** Create, Filter, Group, Nillable, Sort **Description** The template ID used for generating the quote document.
`Status`	**Type** Picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The status of the quote document. Possible values are: • `Completed` • `Failed` • `Generating`

Field	Details
	• `In Progress` • `None` • `Queued` The default value is `None`.

Quote and Order Capture Tooling API Objects

Tooling API exposes metadata used in developer tooling that you can access through REST or SOAP. Tooling API's SOQL capabilities for many metadata types allow you to retrieve smaller pieces of metadata.

TransactionProcessingType
Represents the settings to configure the processing constraints for a request.. This object is available in API version 63.0 and later.

TransactionProcessingType

Represents the settings to configure the processing constraints for a request.. This object is available in API version 63.0 and later.

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.

Supported SOAP API Calls

`create(), describeSObjects(), query(), retrieve()`

Supported REST API Methods

`GET, HEAD, POST, Query`

Fields

Field	Details
`Description`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort **Description** The description of the transaction processing configuration to help Salesforce admins with configuration in their orgs.
`DeveloperName`	**Type** string

Field	Details
	Properties Create, Filter, Group, Sort **Description** Required. The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Label is **Record Type Name**.
`Language`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort **Description** The language of the TransactionProcessingType object. Possible values are: • `da`—Danish • `de`—German • `en_US`—English • `es`—Spanish • `es_MX`—Spanish (Mexico) • `fi`—Finnish • `fr`—French • `it`—Italian • `ja`—Japanese • `ko`—Korean • `nl_NL`—Dutch • `no`—Norwegian • `pt_BR`—Portuguese (Brazil) • `ru`—Russian • `sv`—Swedish • `th`—Thai • `zh_CN`—Chinese (Simplified) • `zh_TW`—Chinese (Traditional)
`MasterLabel`	**Type** string **Properties** Create, Filter, Group, Sort

Field	Details
	Description The label for the TransactionProcessingType object.
`RuleEngine`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort **Description** The rule engine to be used for processing rules. Valid values are: • `AdvancedConfigurator` • `StandardConfigurator`
`SaveType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort **Description** Specifies how the transaction results are processed when saved for Salesforce administrators to adjust the user experience as desired. Valid value is: • `Standard`

Quote and Order Capture Platform Event

Use the QuoteSaveEvent event to notify subscribers after saving of a quote is processed.

PlaceOrderCompletedEvent
Notifies subscribers of an order being created or updated by invoking the Place Order API or the Place Sales Transaction API. This object is available in API version 63.0 and later.

QuoteSaveEvent
Notifies subscribers that the process started by the /actions/standard/quotesaveevent request is complete. If the process is successful, use this event to learn about the updated quote. If the request isn't successful, use this event to learn about the errors and how to fix them. This object is available in API version 60.0 and later.

EDITIONS

Available in: Lightning Experience

Available in all Salesforce orgs where the admin settings for products related to the use case are enabled. The Salesforce org must have a Revenue Cloud or Subscription Management license.

PlaceOrderCompletedEvent

Notifies subscribers of an order being created or updated by invoking the Place Order API or the Place Sales Transaction API. This object is available in API version 63.0 and later.

Supported Calls

`describeSObjects()`

Supported Subscribers

Subscriber	Supported?
Apex Triggers	✔
Flows	✔
Processes	✔
Pub/Sub API	✔
Streaming API (CometD)	✔

Subscription Channel

`/event/PlaceOrderCompletedEvent`

Event Delivery Allocation Enforced

Yes

Fields

Field	Details
`AppUsageTypes`	**Type** string **Properties** Nillable **Description** Tag that represents the application that's using the order and determines how an order is processed. For example, the `AppUsageTypes` field value for Revenue Cloud orders is `RevenueLifecycleManagement`.
`CorrelationIdentifier`	**Type** string **Properties** Nillable **Description** Reserved for future use.
`EventUuid`	**Type** string

Field	Details
	Properties Nillable **Description** A universally unique identifier (UUID) that identifies a platform event message.
`HasErrors`	**Type** boolean **Properties** Defaulted on create **Description** Indicates whether errors occurred when creating or updating the order (`true`) or not (`false`). The default value is `false`.
`OrderId`	**Type** reference **Properties** Nillable **Description** ID of the order record. This field is a relationship field. **Relationship Name** Order **Refers To** Order
`ReplayId`	**Type** string **Properties** Nillable **Description** Represents an ID value that is populated by the system and refers to the position of the event in the event stream. Replay ID values aren't guaranteed to be contiguous for consecutive events. A subscriber can store a replay ID value and use it on resubscription to retrieve missed events that are within the retention window.
`RequestIdentifier`	**Type** string **Properties** Nillable

Field	Details
	Description ID of the request that triggered the event.

QuoteSaveEvent

Notifies subscribers that the process started by the /actions/standard/quotesaveevent request is complete. If the process is successful, use this event to learn about the updated quote. If the request isn't successful, use this event to learn about the errors and how to fix them. This object is available in API version 60.0 and later.

Supported Calls

`describeSObjects()`

Supported Subscribers

Subscriber	Supported?
Apex Triggers	✔
Flows	✔
Processes	✔
Streaming API (CometD)	✔

Streaming API Subscription Channel

`/event/QuoteSaveEvent`

Special Access Rules

This object is available in orgs with Subscription Management or Revenue Cloud.

Fields

Field	Details
`CorrelationIdentifier`	**Type** string **Properties** Nillable **Description** Reserved for future use.

Field	Details
EventUuid	**Type** string **Properties** Nillable **Description** A universally unique identifier (UUID) that identifies a platform event message.
HasErrors	**Type** boolean **Properties** Defaulted on create **Description** The default value is false. Possible values are: • false • true
QuoteId	**Type** reference **Properties** Nillable **Description** The ID of the quote associated with this event. This field is a relationship field.
ReplayId	**Type** string **Properties** Nillable **Description** Represents an ID value that is populated by the system and refers to the position of the event in the event stream. Replay ID values aren't guaranteed to be contiguous for consecutive events. A subscriber can store a replay ID value and use it on resubscription to retrieve missed events that are within the retention window.
RequestIdentifier	**Type** string **Properties** Nillable **Description** The unique ID returned in the /quotesaveevent response. Use this ID to identify the event for the specific request.

Quote and Order Capture Business APIs

Use the Quote and Order Capture Business APIs to fetch instant pricing data on a quote or an order, to create a quote, or to create an order.

This table lists the available Quote and Order Capture resources.

Resource	Description
`/industries/cpq/quotes/actions/get-instant-price` (POST)	Fetch instant pricing data on the quote or order line data grid and associated summary component. It offers capabilities to either create a context or update the existing one based on the provided context ID.
`/commerce/sales-orders/actions/place` (POST)	Place orders with integrated pricing, configuration, and validation, and manage them throughout their entire lifecycle. Additionally, update an order or insert order items.
`/commerce/quotes/actions/place` (POST)	Create a quote to discover and price products and services. Additionally, insert, update, or delete a quote line item.
`/connect/revenue-management/sales-transaction-contexts/`***`resourceId`***`/actions/ramp-deal-create` (POST)	Create a ramp deal for a customer on a product. Sales reps can use ramp deals to provide yearly deals to a customer, resulting in long-term revenue and customer relationship. A customer can create, update, or view multiple segments of periods for their subscription term with different attributes for each segment.
`/connect/revenue-management/sales-transaction-contexts/`***`resourceId`***`/actions/ramp-deal-update` (POST)	Modify a ramp deal in scenarios where a segment has updates such as quantity, discount, or date change.
`/connect/revenue-management/sales-transaction-contexts/`***`resourceId`***`/actions/ramp-deal-view` (GET)	View a ramp deal related to a quote line item or an order item.
`/connect/revenue-management/sales-transaction-contexts/`***`resourceId`***`/actions/ramp-deal-delete` (POST)	Delete a ramp deal to convert a ramped product to include a single quote line item or order item.

Resource	Description
`/connect/rev/sales-transaction/actions/place` (POST)	Create a sales transaction, such as an order or a quote, with integrated pricing and configuration. Additionally, update an order or a quote, and insert and delete order or quote line items to calculate the estimated tax.

Resources
Learn more about the available Quote and Order Capture resources.

Request Bodies
Learn more about the available API request bodies.

Response Bodies
Learn more about the available response bodies.

SEE ALSO:

Connect REST API Developer Guide: Introduction

Resources

Learn more about the available Quote and Order Capture resources.

Instant Pricing (POST)
Fetch instant pricing data on the quote or order line data grid and associated summary component. It offers capabilities to either create a context or update the existing one based on the provided context ID.

Place Order (POST)
Place orders with integrated pricing, configuration, and validation, and manage them throughout their entire lifecycle. Additionally, update an order or insert order items.

Place Quote (POST)
Create a quote to discover and price products and services. Additionally, insert, update, or delete a quote line item.

Place Sales Transaction (POST)
Create a sales transaction, such as an order or a quote, with integrated pricing and configuration. Additionally, update an order or a quote, and insert and delete order or quote line items to calculate the estimated tax.

Create Ramp Deal (POST)
Create a ramp deal for a customer on a product. Sales reps can use ramp deals to provide yearly deals to a customer, resulting in long-term revenue and customer relationship. A customer can create, update, or view multiple segments of periods for their subscription term with different attributes for each segment.

Delete Ramp Deal (POST)
Delete a ramp deal to convert a ramped product to include a single quote line item or order item.

Update Ramp Deal (POST)

Modify a ramp deal in scenarios where a segment has updates such as quantity, discount, or date change.

View Ramp Deal (GET)

View a ramp deal related to a quote line item or an order item.

Instant Pricing (POST)

Fetch instant pricing data on the quote or order line data grid and associated summary component. It offers capabilities to either create a context or update the existing one based on the provided context ID.

You can also group quote line items or order items based on location, work types, or departments, if groups are enabled for your org. Groups provide a visualization of the products to view large quotes.

Resource

```
/industries/cpq/quotes/actions/get-instant-price
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/industries/cpq/quotes/actions/get-instant-price
```

Available version

60.0

Requires Chatter

No

HTTP methods

POST

Request body for POST

JSON example

```
{
  "correlationId": "1234567",
  "contextId": "",
  "records": [
    {
      "referenceId": "refQuote",
      "record": {
        "attributes": {
          "type": "Quote",
          "method": "POST"
        },
        "Name": "Test Quote Proration Pricing",
        "OpportunityId": "006xx000001a4ISAAY",
        "Pricebook2Id": "01sxx0000005ptpAAA"
      }
    },
    {
      "referenceId": "refQuoteLine",
      "record": {
        "attributes": {
          "type": "QuoteLineItem",
          "method": "POST"
        },
```

```
        "QuoteId": "refQuote",
        "PricebookEntryId": "01uxx0000008zHmAAI",
        "Product2Id": "01txx0000006jmWAAQ",
        "Quantity": 2,
        "UnitPrice": 25,
        "StartDate": "2022-09-28",
        "EndDate": "2028-09-27",
        "PeriodBoundary": "ANNIVERSARY",
        "BillingFrequency": "ANNUAL"
      }
    }
  ]
}
```

This example shows a sample request to specify grouping of lines based on criteria.

```
{
    "contextId": "",
    "correlationId": "",
    "records": [
        {
            "referenceId": "0Q0xx0000004DOSCA2",
            "record": {
                "attributes": {
                    "type": "Quote",
                    "method": "PUT"
                }
            }
        },
        {
            "referenceId": "0QLxx0000004F3gGAE",
            "record": {
                "attributes": {
                    "type": "QuoteLineItem",
                    "method": "PUT"
                },
                "Quantity": 5
            }
        }, {
            "referenceId": "0QLxx0000004F3hGAE",
            "record": {
                "attributes": {
                    "type": "QuoteLineItem",
                    "method": "PUT"
                },
                "Quantity": 2
            }
        },
        {
            "referenceId": "GroupId1",
            "record": {
                "attributes": {
                    "type": "QuoteLineGroup",
                    "method": "POST",
                    "action": "GroupBy",
```

```
                    "criteria": {
                        "Quantity": 5
                    }
                },
                "Name": "record"
            }
        }, {
            "referenceId": "GroupId2",
            "record": {
                "attributes": {
                    "type": "QuoteLineGroup",
                    "method": "POST",
                    "action": "GroupBy",
                    "criteria": {
                        "Quantity": 2
                    }
                },
                "Name": "record1",
            }
        }
    ]
}
```

This example shows a sample request for the initial grouping of the quote with the quote lines assigned to the first group.

```
{
    "contextId": "",
    "correlationId": "",
    "records": [
        {
            "referenceId": "0Q0xx0000004CAgCAM",
            "record": {
                "attributes": {
                    "type": "Quote",
                    "method": "PUT"
                }
            }
        },
        {
            "referenceId": "GroupId1",
            "record": {
                "attributes": {
                    "type": "QuoteLineGroup",
                    "method": "POST",
                    "action": "GroupAll"
                },
                "Name": "sample"
            }
        }
    ]
}
```

This example shows a sample request to ungroup a quote but retain the quote lines.

```
{
    "contextId": "",
    "correlationId": "",
    "records": [
        {
            "referenceId": "0Q0xx0000004CAgCAM",
            "record": {
                "attributes": {
                    "type": "Quote",
                    "method": "PUT"
                }
            }
        },
         {
            "referenceId": "0QLxx0000004CBYGA2",
            "record": {
                "attributes": {
                    "type": "QuoteLineItem",
                    "method": "PUT"
                },
                "Quantity": 2,
                "QuoteLineGroupId": null
            }
        },
        {
            "referenceId": "GroupId1",
            "record": {
                "attributes": {
                    "type": "QuoteLineGroup",
                    "method": "DELETE",
                    "action": "Ungroup"
                }
            }
        }
    ]
}
```

This example shows a sample request to create a new group.

```
{
    "contextId": "",
    "correlationId": "",
    "records": [
        {
            "referenceId": "0Q0xx0000004CAgCAM",
            "record": {
                "attributes": {
                    "type": "Quote",
                    "method": "PUT"
                }
            }
        },
        {
```

```
            "referenceId": "GroupId1",
            "record": {
                "attributes": {
                    "type": "QuoteLineGroup",
                    "method": "POST"
                },
                "Name": "sample"
            }
        }
    ]
}
```

This example shows a sample request to delete a group.

```
{
    "contextId": "",
    "correlationId": "",
    "records": [
        {
            "referenceId": "0Q0xx0000004CAgCAM",
            "record": {
                "attributes": {
                    "type": "Quote",
                    "method": "PUT"
                }
            }
        },
        {
            "referenceId": "GroupId1",
            "record": {
                "attributes": {
                    "type": "QuoteLineGroup",
                    "method": "DELETE",
                    "action": "DeleteGroup"
                },
                "Name": "sample"
            }
        }
    ]
}
```

This example shows a sample request to move a group.

```
{
    "contextId": "",
    "correlationId": "",
    "records": [
        {
            "referenceId": "0Q0xx0000004CAgCAM",
            "record": {
                "attributes": {
                    "type": "Quote",
                    "method": "PUT"
                }
            }
```

```
        },
        {
            "referenceId": "0QLxx0000004CBYGA2",
            "record": {
                "attributes": {
                    "type": "QuoteLineItem",
                    "method": "PUT"
                },
                "Quantity": 2
                "QuoteLineGroupId": "{@GroupId2}"
            }
        },
    ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`contextId`	String	The ID generated by the context service. If no context ID is provided, a new context is created.	Optional	59.0
`correlationId`	String	Client-generated ID for tracking multiple related API requests.	Optional	59.0
`records`	Object with Reference Input[]	List of pricing data to be fetched.	Required	59.0

Response body for POST

Instant Pricing

Place Order (POST)

Place orders with integrated pricing, configuration, and validation, and manage them throughout their entire lifecycle. Additionally, update an order or insert order items.

You can also group order items based on location, work types, or departments, if groups are enabled for your org. Groups provide a visualization of the products to view large quotes.

This API supports a maximum of 300 transaction line items. To place sales transactions that exceed this limit, use the Place Sales Transaction API. See Place Sales Transaction (POST).

Special Access Rules

You need the PlaceOrder API permission set to use this API.

Resource

```
/commerce/sales-orders/actions/place
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/commerce/sales-orders/actions/place
```

Available version

60.0

HTTP methods

POST

Request body for POST

JSON example

```
{
  "pricingPref": "System",
  "configurationInput": "RunAndAllowErrors",
  "configurationOptions": {
    "validateProductCatalog": true,
    "validateAmendRenewCancel": true,
    "executeConfigurationRules": true,
    "addDefaultConfiguration": true
  },
  "graph": {
    "graphId": "graphId",
    "records": [
      {
        "referenceId": "refOrder",
        "record": {
          "attributes": {
            "type": "Order",
            "method": "POST",
            "Id": "POST"
          }
        }
      },
      {
         "referenceId":"refOrderItem",
         "record":{
         "attributes":{
             "type":"OrderItem",
             "method":"POST"
            },
            "OrderId":"@{refOrder.id}",
            "OrderActionId":"@{refOrderAction.id}",
            "ListPrice":"144.99",
            "Quantity":3,
            "PricebookEntryId":"01uxx0000008yXPAAY",
            "Product2Id":"01txx0000006i2UAAQ",
            "UnitPrice":"199.49"
       }
      }
    ]
  }
}
```

This example shows a sample request to define grouping of order items.

```
{
    "pricingPref": "system",
```

```
    "graph": {
        "graphId": "placeOrder",
        "records": [
            {
                "referenceId": "refOrder",
                "record": {
                    "attributes": {
                        "type": "Order",
                        "method": "PATCH",
                        "id": "801xx000003GZ9bAAG"
                    }
                }
            },
            {
                "referenceId": "refOlg1",
                "record": {
                    "attributes": {
                        "type": "OrderItemGroup",
                        "method": "POST"
                    },
                    "Name": "New Group",
                    "OrderId": "@{refOrder.id}"
                }
            }
        ]
    }
}
```

This example shows a sample request to ungroup order items.

```
{
    "pricingPref": "system",
    "graph": {
        "graphId": "placeOrder",
        "records": [
            {
                "referenceId": "refOrder",
                "record": {
                    "attributes": {
                        "type": "Order",
                        "method": "PATCH",
                        "id": "refOrder"
                    }
                }
            },
            {
                "referenceId": "refOlg1",
                "record": {
                    "attributes": {
                        "type": "OrderItemGroup",
                        "method": "DELETE",
                        "id": "refOlg1",
                        "action": "Ungroup"
                    }
                }
```

```
            }
        ]
    }
}
```

This example shows a sample request to create a new group.

```
{
    "pricingPref": "system",
    "graph": {
        "graphId": "placeOrder",
        "records": [
            {
                "referenceId": "refOrder",
                "record": {
                    "attributes": {
                        "type": "Order",
                        "method": "PATCH",
                        "id": "refOrder"
                    }
                }
            },
            {
                "referenceId": "refOlg",
                "record": {
                    "attributes": {
                        "type": "OrderItemGroup",
                        "method": "POST"
                    },
                    "Name": "New Group",
                    "OrderId": "@{refOrder.id}"
                }
            }
        ]
    }
}
```

This example shows a sample request to delete a group.

```
{
    "contextId": "",
    "correlationId": "",
    "records": [
        {
            "referenceId": "refOrder",
            "record": {
                "attributes": {
                    "type": "Order",
                    "method": "PATCH",
                    "id": "refOrder"
                }
            }
        },
        {
            "referenceId": "refOlg",
```

```
            "record": {
                "attributes": {
                    "type": "OrderItemGroup",
                    "method": "DELETE",
                    "id": "refOlg",
                    "action": "DeleteGroup"
                }
            }
        }
    ]
}
```

This example shows a sample request to group order items based on criteria.

```
{
    "pricingPref": "system",
    "graph": {
        "graphId": "placeOrder",
        "records": [
            {
                "referenceId": "refOrder",
                "record": {
                    "attributes": {
                        "type": "Order",
                        "method": "PATCH",
                        "id": "refOrder"
                    }
                }
            },
            {
                "referenceId": "g0",
                "record": {
                    "attributes": {
                        "type": "OrderItemGroup",
                        "method": "POST",
                        "action": "GroupBy",
                        "criteria": {
                            "BillingFrequency2": null
                        }
                    },
                    "Name": "Billing Frequency: ",
                    "OrderId": "@{refOrder.id}"
                }
            },
            {
                "referenceId": "g1",
                "record": {
                    "attributes": {
                        "type": "OrderItemGroup",
                        "method": "POST",
                        "action": "GroupBy",
                        "criteria": {
                            "BillingFrequency2": "Monthly"
                        }
                    },
```

```
                    "Name": "Billing Frequency: Monthly",
                    "OrderId": "@{refOrder.id}"
                }
            }
        ]
    }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`catalogRatesPref`	String	Rate card entries defined in the catalog that must be fetched for order items with usage-based pricing during the order creation process. Valid values are: • `Fetch`—Retrieves the rate card entries defined in the catalog for order items during the order creation process. • `Skip`—Skips the retrieval of rate card entries for order items during the order creation process. The default value is `Skip`. This property is available when the Usage-Based Selling feature is enabled.	Optional	62.0
`configurationInput`	String	Configuration input for the place order process. Valid values are: • `RunAndAllowErrors`—Specifies to run the configuration and to proceed order ingestion upon encountering any configuration errors. • `RunAndBlockErrors`—Specifies to run configuration and to block order ingestion upon encountering any configuration errors. • `Skip`—Specifies to skip configuration. The default value is `RunAndBlockErrors`.	Optional	60.0
`configurationOptions`	Configuration Options Input[]	Configuration options during the ingestion process.	Optional	60.0
`graph`	Object Graph Input	The sObject graph of the order payload to be ingested. You can perform create,	Required	60.0

Name	Type	Description	Required or Optional	Available Version
		update, or delete operations on objects from the Sales Transaction context definition by using this property. Additionally, perform create, update, or delete operations on custom objects and fields in your extended context definition.		
`pricingPref`	String	Pricing preference during the create order process. Valid values are: • `Force`—Specifies to force pricing during the order ingestion process. • `Skip`—Specifies to skip pricing during the order ingestion process. • `System`—Specifies the system to determine whether a pricing calculation is required. The default value is `System`.	Optional	60.0

Response body for POST

Place Order

Place Quote (POST)

Create a quote to discover and price products and services. Additionally, insert, update, or delete a quote line item.

You can also group quote line items based on location, work types, or departments, if groups are enabled for your org. Groups provide a visualization of the products to view large quotes.

This API supports a maximum of 300 transaction line items. To place sales transactions that exceed this limit, use the Place Sales Transaction API. See Place Sales Transaction (POST).

Special Access Rules

You need the Create on Quotes user permission to create quotes.

Resource

```
/commerce/quotes/actions/place
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/commerce/quotes/actions/place
```

Available version

60.0

HTTP methods

POST

Request body for POST

JSON example

This example shows a sample request to create a quote.

```
{
  "pricingPref": "System",
  "configurationInput": "RunAndAllowErrors",
  "configurationOptions": {
    "validateProductCatalog": true,
    "validateAmendRenewCancel": true,
    "executeConfigurationRules": true,
    "addDefaultConfiguration": true
  },  "graph": {
    "graphId": "createQuote",
    "records": [{
        "referenceId": "refQuote",
        "record": {
          "attributes": {
            "type": "Quote",
            "method": "POST"
          },
          "opportunityId": "---",
          "quoteProp1": "value1",
          "quoteProp2": "value2"
        }
      },
      {
        "referenceId": "refQuoteLineItem1",
        "record": {
          "attributes": {
            "type": "QuoteLineItem",
            "method": "POST"
          },
          "QuoteLineItemProp1": "value1",
          "QuoteLineItemProp2": "value2"
        }
      },
      {
      "referenceId": "refQuoteLineItemAttribute",
      "record": {
        "attributes": {
              "type": "QuoteLineItemAttribute",
              "method": "POST"
       },
       "QuoteLineItemId": "@{refQuoteItem1.id}",
       "AttributeDefinitionId": "0tjxx0000000001AAA"
      }
}
    ]
  }
}
```

This example shows a sample request to insert, update, or delete a quote line item.

```
{
  "pricingPref": "System",
  "configurationInput": "skip",
  "graph": {
    "graphId": "updateQuote",
    "records": [
      {
        "referenceId": "refQuote",
        "record": {
          "attributes": {
            "type": "Quote",
            "method": "PATCH",
            "id": "0Q0xx0000004E2mCAE"
          },
          "Name": "Quote_Acme"
        }
      },
      {
        "referenceId": "refQuoteLineItemToCreate1",
        "record": {
          "attributes": {
            "type": "QuoteLineItem",
            "method": "POST"
          },
          "QuoteId": "0Q0xx0000004E2mCAE",
          "PricebookEntryId": "01uxx0000008yXPAAY",
          "Product2Id": "01txx0000006i2UAAQ",
          "Quantity": 2.0,
          "UnitPrice": 800.0,
          "PeriodBoundary": "Anniversary",
          "BillingFrequency": "Monthly",
          "StartDate": "2024-03-11"
        }
      },
      {
        "referenceId": "refQuoteLineItemToPatch2",
        "record": {
          "attributes": {
            "type": "QuoteLineItem",
            "method": "PATCH",
            "id": "0Q0xx0000004E2mCAE"
          },
          "Quantity": 2.0,
          "UnitPrice": 600.0
        }
      },
      {
        "referenceId": "refQuoteLineItemToDelete3",
        "record": {
          "attributes": {
            "type": "QuoteLineItem",
            "method": "DELETE",
            "id": "0Q0xx0000004E2mYLK"
```

```
            }
          }
        }
      ]
    }
}
```

This example shows a sample request to define grouping of quote line items.

```
{
   "pricingPref": "Force",
   "configurationInput": "skip",
   "graph": {
      "graphId": "groupLines",
      "records": [
         {
            "referenceId": "refQuote",
            "record": {
               "attributes": {
                  "type": "Quote",
                  "method": "PATCH",
                  "id":"0Q0xx0000004C99CAE"
               },
               "Name": "From Place Quote API"
            }
         },
         {
            "referenceId": "refQlg1",
            "record": {
               "attributes": {
                  "type": "QuoteLineGroup",
                  "method": "POST",
                  "action": "GroupBy",
                  "criteria": {
                    "Quantity": 1
                  }
               },
               "Name": "From Place Quote API Group",
               "QuoteId": "@{refQuote.id}"
            }
         },
         {
            "referenceId": "refQuoteItem1",
            "record": {
               "attributes": {
                  "type": "QuoteLineItem",
                  "method": "PATCH",
                  "id":"0QLxx0000004DJcGAM"
               },
               "QuoteLineGroupId": "@{refQlg1.id}",
               "Quantity": 1
            }
         }
      ]
```

```
    }
}
```

This example shows a sample request for the initial grouping of the quote with the quote lines assigned to the first group.

```
{
  "pricingPref": "Force",
  "graph": {
    "graphId": "test",
    "records": [
      {
        "referenceId": "refQuote",
        "record": {
          "attributes": {
            "type": "Quote",
            "method": "PATCH",
            "id": "0Q0xx0000004CAmCAM"
          }
        }
      },
      {
        "referenceId": "refQlg1",
        "record": {
          "attributes": {
            "type": "QuoteLineGroup",
            "method": "POST",
            "action": "GroupAll"
          },
          "Name": "From PQ API Group",
          "QuoteId": "@{refQuote.id}"
        }
      }
    ]
  }
}
```

This example shows a sample request to ungroup a quote but retain the quote lines.

```
{
   "pricingPref": "Force",
   "configurationInput": "skip",
   "graph": {
      "graphId": "test",
      "records": [
         {
            "referenceId": "refQuote",
            "record": {
               "attributes": {
                  "type": "Quote",
                  "method": "PATCH",
                  "id":"0Q0xx0000004C99CAE"
               },
               "Name": "From Place Quote API"
            }
         },
```

```
            {
               "referenceId": "refQlg1",
               "record": {
                  "attributes": {
                     "type": "QuoteLineGroup",
                     "method": "DELETE",
                     "id": "{GroupId}",
                     "action": "Ungroup"
                  }
               }
            }
         ]
      }
}
```

This example shows a sample request to create a new group.

```
{
   "pricingPref": "Force",
   "configurationInput": "skip",
   "graph": {
      "graphId": "test",
      "records": [
         {
            "referenceId": "refQuote",
            "record": {
               "attributes": {
                  "type": "Quote",
                  "method": "PATCH",
                  "id":"0Q0xx0000004C99CAE"
               },
               "Name": "From Place Quote API"
            }
         },
         {
            "referenceId": "refQlg1",
            "record": {
               "attributes": {
                  "type": "QuoteLineGroup",
                  "method": "POST"
               },
               "Name": "From PQ API Group",
               "QuoteId": "@{refQuote.id}"
            }
         }
      ]
   }
}
```

This example shows a sample request to delete a group.

```
{
   "pricingPref": "Force",
   "configurationInput": "skip",
   "graph": {
```

```
      "graphId": "test",
      "records": [
         {
            "referenceId": "refQuote",
            "record": {
               "attributes": {
                  "type": "Quote",
                  "method": "PATCH",
                  "id":"0Q0xx0000004C99CAE"
               },
               "Name": "From Place Quote API"
            }
         },
         {
            "referenceId": "refQlg1",
            "record": {
               "attributes": {
                  "type": "QuoteLineGroup",
                  "method": "DELETE",
                  "id": "{GroupId}",
                  "action": "DeleteGroup"
               }
            }
         }
      ]
   }
}
```

This example shows a sample request to move a group.

```
{
   "pricingPref": "Force",
   "configurationInput": "skip",
   "graph": {
      "graphId": "test",
      "records": [
         {
            "referenceId": "refQuote",
            "record": {
               "attributes": {
                  "type": "Quote",
                  "method": "PATCH",
                  "id":"0Q0xx0000004C99CAE"
               },
               "Name": "From PlaceQuote Api"
            }
         },
        {
            "referenceId": "0QLxx0000004CBYGA2",
            "record": {
                "attributes": {
                    "type": "QuoteLineItem",
                    "method": "PATCH"
                    "id": "0QLxx0000004CBYGA2"
                },
```

```
                "Quantity": 2,
                "QuoteLineGroupId": "@{GroupId2}"
            }
        ]
    }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`catalogRatesPref`	String	Rate card entries defined in the catalog that must be fetched for quote line items with usage-based pricing during the quote creation process. Valid values are: • `Fetch`—Retrieves the rate card entries defined in the catalog for quote line items during the quote creation process. • `Skip`—Skips the retrieval of rate card entries for quote line items during the quote creation process. The default value is `Skip`. This property is available when the Usage-Based Selling feature is enabled.	Optional	62.0
`configurationInput`	String	Configuration input for the place quote process. Valid values are: • `RunAndAllowErrors` • `RunAndBlockErrors` • `Skip` The default value is `RunAndBlockErrors`.	Optional	60.0
`configurationOptions`	Configuration Options Input	Configuration options during the ingestion process.	Optional	60.0
`graph`	Object Graph Input	The sObject graph representing the quote structure. You can perform create, update, or delete operations on objects from the Sales Transaction context definition by using this property. Additionally, perform create, update, or delete operations on custom objects and fields in your extended context definition.	Required	60.0

Name	Type	Description	Required or Optional	Available Version
`pricingPref`	String	Pricing preference during the quote process. Valid values are: • `Force` • `Skip` • `System` The default value is `System`.	Optional	60.0

Response body for POST

Place Quote

Place Sales Transaction (POST)

Create a sales transaction, such as an order or a quote, with integrated pricing and configuration. Additionally, update an order or a quote, and insert and delete order or quote line items to calculate the estimated tax.

You can also group order or quote line items based on location, work types, or departments, if groups are enabled for your org. Groups provide a visualization of the products to view large quotes. You can add up to 1000 quote line items for a quote, and 1000 order products for an order.

Resource

```
/connect/rev/sales-transaction/actions/place
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/rev/sales-transaction/actions/place
```

Available version

63.0

HTTP methods

POST

Request body for POST

JSON example

This is a sample request to create a sales transaction for a quote.

```
{
  "pricingPref": "system",
  "catalogRatesPref" "skip",
  "configurationPref": {
    "configurationMethod": "Skip",
    "configurationOptions": {
      "validateProductCatalog": true,
      "validateAmendRenewCancel": true,
      "executeConfigurationRules": true,
      "addDefaultConfiguration": true
    }
  },
```

```
  "contextDetails": {
    "contextId": "e055bb18-d4e8-41c3-881e-0132b9561708"
  },
  "graph": {
    "graphId": "createQuote",
    "records": [
      {
        "referenceId": "refQuote",
        "record": {
          "attributes": {
            "method": "POST",
            "type": "Quote"
          },
          "Name": "Quote_Acme",
          "Pricebook2Id": "01sDU000000JvhbYAC"
        }
      },
      {
        "referenceId": "refQuoteLine0",
        "record": {
          "attributes": {
            "type": "QuoteLineItem",
            "method": "POST"
          },
          "QuoteId": "@{refQuote.id}",
          "Product2Id": "01tDU000000F7b8YAC",
          "PricebookEntryId": "01uDU000000fxt2YAA",
          "UnitPrice": 100,
          "Quantity": "1",
          "StartDate": "2024-10-29",
          "EndDate": "2025-03-01",
          "PeriodBoundary": "Anniversary"
        }
      }
    ]
  }
}
```

This is a sample request to insert, update, or delete a quote line item.

```
{
  {
  "pricingPref": "system",
  "catalogRatesPref": "skip",
  "configurationPref": {
    "configurationMethod": "Skip",
    "configurationOptions": {
      "validateProductCatalog": true,
      "validateAmendRenewCancel": true,
      "executeConfigurationRules": true,
      "addDefaultConfiguration": true
    }
  },
  "contextDetails": {
    "contextId": "e055bb18-d4e8-41c3-881e-0132b9561708"
```

```
  },
  "graph": {
    "graphId": "updateQuote",
    "records": [
      {
        "referenceId": "refQuote",
        "record": {
          "attributes": {
            "method": "PATCH",
            "type": "Quote",
            "id": "801xx000003GZ9bAAG"
          }
        }
      },
      {
        "referenceId": "refQuoteLine0",
        "record": {
          "attributes": {
            "type": "QuoteLineItem",
            "method": "PATCH",
            "id": "402xx000003KY5vJGH"
          },
          "Quantity": "5"
        }
      }
    ]
  }
}
```

This is a sample request to define grouping of quote line items.

```
{
  "pricingPref": "Force",
  "catalogRatesPref" "skip",
  "configurationPref": {
    "configurationMethod": "Skip",
    "configurationOptions": {
      "validateProductCatalog": true,
      "validateAmendRenewCancel": true,
      "executeConfigurationRules": true,
      "addDefaultConfiguration": true
    }
  },
  "contextDetails": {
    "contextId": "e055bb18-d4e8-41c3-881e-0132b9561708"
  },
  "graph": {
    "graphId": "groupQuoteLines",
    "records": [
      {
        "referenceId": "refQuote",
        "record": {
          "attributes": {
            "method": "PATCH",
            "type": "Quote",
```

```
            "id": "801xx000003GZ9bAAG"
          }
        }
      },
      {
        "referenceId": "refQuoteLine0",
        "record": {
          "attributes": {
            "type": "QuoteLineItem",
            "method": "PATCH",
            "id": "402xx000003KY5vJGH"
          },
          "QuoteLineGroupId": "@{refQuote.id}"
        }
      }
    ]
  }
}
```

This is s a sample request for the initial grouping of the quote with all the quote lines assigned to the first group.

```
{
  "pricingPref": "Force",
  "catalogRatesPref" "skip",
  "graph": {
    "graphId": "groupQuote",
    "records": [
      {
        "referenceId": "refQuote",
        "record": {
          "attributes": {
            "type": "Quote",
            "method": "PATCH",
            "id": "0Q0xx0000004CAmCAM"
          }
        }
      },
      {
        "referenceId": "refQlg1",
        "record": {
          "attributes": {
            "type": "QuoteLineGroup",
            "method": "POST",
            "action": "GroupAll"
          },
          "Name": "From PST API Group",
          "QuoteId": "@{refQuote.id}"
        }
      }
    ]
  }
}
```

This is a sample request to ungroup a quote but retain the quote lines.

```
{
   "catalogRatesPref" "skip",
   "pricingPref": "Force",
   "graph": {
      "graphId": "ungroupQuote",
      "records": [
         {
            "referenceId": "refQuote",
            "record": {
               "attributes": {
                  "type": "Quote",
                  "method": "PATCH",
                  "id":"0Q0xx0000004C99CAE"
               },
               "Name": "Grouped Quote with PST API"
            }
         },
         {
            "referenceId": "refQlg1",
            "record": {
               "attributes": {
                  "type": "QuoteLineGroup",
                  "method": "DELETE",
                  "id": "402xx000003KY5vJGH",
                  "action": "Ungroup"
               }
            }
         }
      ]
   }
}
```

This is a sample request to create a new group.

```
{
   "catalogRatesPref" "skip",
   "pricingPref": "Force",
   "graph": {
      "graphId": "createGroup",
      "records": [
         {
            "referenceId": "refQuote",
            "record": {
               "attributes": {
                  "type": "Quote",
                  "method": "PATCH",
                  "id":"0Q0xx0000004C99CAE"
               },
               "Name": "Grouped Quote with PST API"
            }
         },
         {
            "referenceId": "refQlg1",
```

```
            "record": {
               "attributes": {
                  "type": "QuoteLineGroup",
                  "method": "POST"
               },
               "Name": "From PQ API Group",
               "QuoteId": "@{refQuote.id}"
            }
         }
      ]
   }
}
```

This example shows a sample request to delete a group.

```
{
   "catalogRatesPref" "skip",
   "pricingPref": "Force",
   "graph": {
      "graphId": "deleteGroup",
      "records": [
         {
            "referenceId": "refQuote",
            "record": {
               "attributes": {
                  "type": "Quote",
                  "method": "PATCH",
                  "id":"0Q0xx0000004C99CAE"
               },
               "Name": "Grouped Quote with PST API"
            }
         },
         {
            "referenceId": "refQlg1",
            "record": {
               "attributes": {
                  "type": "QuoteLineGroup",
                  "method": "DELETE",
                  "id": "402xx000003KY5vJGH",
                  "action": "DeleteGroup"
               }
            }
         }
      ]
   }
}
```

This is a sample request to group order items based on criteria.

```
{
    "catalogRatesPref" "skip",
    "pricingPref": "Force",
    "graph": {
        "graphId": "groupOrderItems",
        "records": [
```

```
            {
                "referenceId": "refOrder",
                "record": {
                    "attributes": {
                        "type": "Order",
                        "method": "PATCH",
                        "id": "0Q0xx0000004C99CAE"
                    }
                }
            },
            {
                "referenceId": "refOlg1",
                "record": {
                    "attributes": {
                        "type": "OrderItemGroup",
                        "method": "POST",
                        "action": "GroupBy",
                        "criteria": {
                            "BillingFrequency2": null
                        }
                    },
                    "Name": "Billing Frequency: ",
                    "OrderId": "@{refOrder.id}"
                }
            },
            {
                "referenceId": "g1",
                "record": {
                    "attributes": {
                        "type": "OrderItemGroup",
                        "method": "POST",
                        "action": "GroupBy",
                        "criteria": {
                            "BillingFrequency2": "Monthly"
                        }
                    },
                    "Name": "Billing Frequency: Monthly",
                    "OrderId": "@{refOrder.id}"
                }
            }
        ]
    }
}
```

This is a sample request to save changes to a ramp deal by using context ID. The context ID is returned by the Ramp Deal APIs. See Create Ramp Deal (POST).

```
{
  "pricingPref": "force",
  "contextDetails": {
    "contextId": "f1c9e3e1c335f7959a88de09d3a867cc2b95e08709b99de8e2edeb8f5039e8ed",

    "scope": "Session"
  },
  "graph": {
```

```
    "graphId": "updateQuote",
    "records": [
      {
        "referenceId": "refQuote",
        "record": {
          "attributes": {
            "type": "Quote",
            "method": "PATCH",
            "id": "0Q0xx0000004DQ4CAM"
          }
        }
      }
    ]
  }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`catalogRatesPref`	String	Rate card entries defined in the catalog that must be fetched for sales items with usage-based pricing during the creation of the sales transaction. Valid values are: • `Fetch`—Retrieves the rate card entries defined in the catalog for sales items during the creation of the sales transaction. • `Skip`—Skips the retrieval of rate card entries for sales items during the creation of the sales transaction. The default value is `Skip`. This property is available when the Usage-Based Selling feature is enabled.	Optional	63.0
`configurationPref`	Configurator Preference Input	Configuration preference during the quote process. These preferences ensure that quotes are defined as per the requirement.	Optional	63.0
`contextDetails`	Context Input	Context details that are created for a sales transaction.	Required if the `graph` property isn't specified.	63.0
`graph`	Object Graph Input	The sObject graph of the sales transaction to be ingested. You can perform create, update, or delete operations on objects from the Sales Transaction context definition by using this property. Additionally, perform create, update, or	Required if the `contextDetails` property isn't specified.	63.0

Name	Type	Description	Required or Optional	Available Version
		delete operations on custom objects and fields in your extended context definition.		
`pricingPref`	String	Pricing preference during the creation of a sales transaction. Valid values are: • `Force`—Specifies to enforce pricing during the creation of sales transactions. • `Skip`—Specifies to skip pricing during the creation of sales transactions. • `System`—Specifies the system to determine whether a pricing calculation is required. The default value is `System`.	Optional	63.0

Response body for POST

Place Sales Transaction

Create Ramp Deal (POST)

Create a ramp deal for a customer on a product. Sales reps can use ramp deals to provide yearly deals to a customer, resulting in long-term revenue and customer relationship. A customer can create, update, or view multiple segments of periods for their subscription term with different attributes for each segment.

This API request creates segments based on the specified input properties such as term, segment type, and trial details. The API response includes the context ID and the updated context object for the sales transaction. You must call the Place Sales Transaction (POST) API by specifying this context ID to apply the ramp deal updates. See Place Sales Transaction (POST) API.

Resource

```
/connect/revenue-management/sales-transaction-contexts/resourceId/actions/ramp-deal-create
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/revenue-management/sales-transaction-contexts/0QLx0000004CFIGAU/actions/ramp-deal-create
```

Available version

62.0

HTTP methods

POST

Path parameter for POST

Name	Type	Description	Required or Optional	Available Version
`resourceId`	String	ID of the quote line item, order item, or context.	Required	62.0

Request body for POST

JSON example

```
{
  "transactionId": "0Q0xx0000004C92CAE",
  "transactionLineId": "0QLxx0000004C9VGAU",
  "subscriptionTerm": 14,
  "subscriptionTermUnit": "MONTHS",
  "trialTerm": 45,
  "trialTermUnit": "DAYS",
  "segmentType": "YEARLY",
  "executionSettings": {
       "executePricing": true,
       "executeConfigRules": false
   }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`execution Settings`	Execution Settings Input[]	Settings to run the pricing or configuration rules.	Optional	62.0
`segmentType`	String	Type of segment that the user wants to create. Valid values are: • `FREE_TRIAL` • `CUSTOM` • `YEARLY`	Required	62.0
`subscription Term`	Integer	Subscription length of the term-defined product.	Required	62.0
`subscription TermUnit`	String	Unit of time for the subscription length. Valid value is: • `MONTHS`	Required	62.0
`transaction Id`	String	ID of the sales transaction that's configured, such as quote or order.	Required	62.0

Name	Type	Description	Required or Optional	Available Version
`transactionLineId`	String	Quote line item ID or order item ID that the price ramp is created for.	Required	62.0
`trialTerm`	Integer	Length of the trial period, if any.	Optional	62.0
`trialTermUnit`	String	Unit of time for the trial period. Valid value is: • `DAYS`	Optional. Required if `trialTerm` property is specified.	62.0

Response body for POST

Ramp Deal Service

Delete Ramp Deal (POST)

Delete a ramp deal to convert a ramped product to include a single quote line item or order item.

This API request deletes the segments related to the product. The API response includes the updated context with the context ID. You must call the Place Sales Transaction (POST) API by specifying this context ID to apply the ramp deal updates. See Place Sales Transaction (POST) API.

Resource

```
/connect/revenue-management/sales-transaction-contexts/resourceId/actions/ramp-deal-delete
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/revenue-management/sales-transaction-contexts/0QLxx0000004CFIGAU/actions/ramp-deal-delete
```

Available version

62.0

HTTP methods

POST

Path parameter for POST

Name	Type	Description	Required or Optional	Available Version
`resourceId`	String	ID of the context.	Required	62.0

Request body for POST

JSON example

```
{
  "rampDealIds": [
    "0Q0xx0000004CDxCAM",
    "0QLxx0000004CSOGA2"
```

```
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`rampDealIds`	String[]	Ramp identifier on the quote line item or order item.	Required	62.0

Response body for POST

Ramp Deal Service

Update Ramp Deal (POST)

Modify a ramp deal in scenarios where a segment has updates such as quantity, discount, or date change.

Update a ramp deal in these scenarios.

- A segment has quantity or discount changes.
- A trial segment or custom segment has a date change. A custom segment is an added or deleted segment. In this scenario, you can update a ramp deal during the initial sale before assetization.

This API request returns the updated context with the context ID. You must call the Place Sales Transaction (POST) API by specifying this context ID to apply the ramp deal updates. See Place Sales Transaction (POST) API.

Resource

```
/connect/revenue-management/sales-transaction-contexts/resourceId/actions/ramp-deal-update
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/revenue-management/sales-transaction-contexts/4f2361-5c98f0f8f0f5e04c6b37e9f1b8a27a3a0b5f1d410e9/actions/ramp-deal-update
```

Available version

62.0

HTTP methods

POST

Path parameter for POST

Name	Type	Description	Required or Optional	Available Version
`resourceId`	String	ID of the context data that's used to build the pricing procedure. Get the context instance ID by invoking the Context Service API. See Context Service (POST).	Required	62.0

Request body for POST

JSON example

```
{
  "executionSettings": {
       "executePricing": true,
       "executeConfigRules": false
  },
  "addedNodes": [
    {
      "contextNodePath": [
       "4f23961a5c98806f89305e064c67b397e93f1bb8a2a7a3a80db506f1d4110ee9", // Context
 ID
        "0Q0xx0000004CPACA2", //Quote or Order ID
        "RandomUUID" // random UUID for Quote Line Item or Order Item ID
      ],
      "contextNode": {
         "Discount": 10,
         "Quantity": 5,
         "ItemSegmentName": "Year 5",
         "StartDate":"2024-09-07T00:00:00.000Z",
         "EndDate":"2024-09-07T00:00:00.000Z"
      }
    }
  ],
  "updatedNodes": [
     {
      "contextNodePath": [
       "4f23961a5c98806f89305e064c67b397e93f1bb8a2a7a3a80db506f1d4110ee9", // Context
 ID
        "0Q0xx0000004CPACA2", //Quote or Order ID
        "0QLxx0000004CfIGAU" // Quote Line ID or Order Line ID to update
      ],
      "contextNode": {
          "Discount": 10,
          "Quantity": 5
      }
    }
  ],
  "deletedNodes": [
    {
      "contextNodePath": [
        "4f23961a5c98806f89305e064c67b397e93f1bb8a2a7a3a80db506f1d4110ee9",
        "0Q0xx0000004CPACA2",
        "0QLxx0000004CfIGAU" // Quote Line Item ID to delete
      ]
    }
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`addedNodes`	Context Node Input[]	Details of the nodes to be added.	Required	62.0
`deletedNodes`	Context Node Input[]	Details of the nodes to be deleted.	Required	62.0
`execution Settings`	Execution Settings Input[]	Settings to run the pricing or configuration rules.	Optional	62.0
`updatedNodes`	Context Node Input[]	Details of the nodes to be updated.	Required	62.0

Response body for POST

Ramp Deal Service

View Ramp Deal (GET)

View a ramp deal related to a quote line item or an order item.

This API request retrieves the segments if the ramp deal already exists.

Resource

```
/connect/revenue-management/sales-transaction-contexts/resourceId/actions/ramp-deal-view
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/revenue-management/sales-transaction-contexts/0Qx000004SS02/actions/ramp-deal-view?transactionId=0Qx000004DXAMtransactionLineId=0Qx000004SS02
```

Available version

62.0

HTTP methods

GET

Path parameter for GET

Name	Type	Description	Required or Optional	Available Version
`resourceId`	String	ID of the quote line item, order item, or context.	Required	62.0

Request parameters for GET

Parameter Name	Type	Description	Required or Optional	Available Version
`transactionId`	String	ID of the quote or order required to hydrate the context and retrieve the quote lines.	Required	62.0
`transaction LineId`	String	ID of the quote or order line required to retrieve the segmented details.	Required	62.0

Response body for GET

Ramp Deal Service

Request Bodies

Learn more about the available API request bodies.

Configuration Options Input
Input representation for the configuration options.

Configurator Preference Input
Input representation of the configuration preference for the place sales transaction request.

Context Input
Input representation of the context that's associated with a sales transaction for a quote or an order.

Context Node Input
Input representation of the details of the context nodes for ramp segments.

Execution Settings Input
Input representation of the execution settings for a ramp deal.

Instant Pricing Input
Input representation to fetch the instant pricing details.

Object Graph Input
Input representation of an sObject with a graph ID.

Object Input Map
Input representation of an sObject record in a key-value map format.

Object with Reference Input
Input representation of a list of records to be inserted or updated. To update a record, specify the record ID.

Place Order Input
Input representation of the request to create or update an order.

Place Quote Input
Input representation of the request to create or update a quote.

Place Sales Transaction Input
Input representation of the details of the request to place a sales transaction, such as a quote or an order.

Create Ramp Deal Input
Input representation of the request to create a ramp deal.

Delete Ramp Deal Input
Input representation of the request to delete a ramp deal.

Update Ramp Deal Input
Input representation of the request to update a ramp deal.

Configuration Options Input

Input representation for the configuration options.

JSON example

```
{
  "configurationOptions": {
    "validateProductCatalog": true,
    "validateAmendRenewCancel": true,
    "executeConfigurationRules": true,
    "addDefaultConfiguration": true
  }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`addDefaultConfiguration`	Boolean	Indicates whether to automatically add default configurations to the order (`true`) or not (`false`).	Optional	60.0
`executeConfigurationRules`	Boolean	Indicates whether the order must adhere to configuration rules during processing (`true`) or bypass them (`false`).	Optional	60.0
`validateAmendRenewCancel`	Boolean	Indicates whether to run validations related to amend, renew, or cancel processes (`true`) or not (`false`).	Optional	60.0
`validateProductCatalog`	Boolean	Indicates whether the order must be validated against the product catalog (`true`) or not (`false`).	Optional	60.0

Configurator Preference Input

Input representation of the configuration preference for the place sales transaction request.

JSON example

```
{
  "configurationPref": {
    "configurationMethod": "Skip",
    "configurationOptions": {
```

```
      "validateProductCatalog": true,
      "validateAmendRenewCancel": true,
      "executeConfigurationRules": true,
      "addDefaultConfiguration": true
    }
  }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`configurationMethod`	String	Configuration method for the place sales transaction request. Valid values are: • `Force`—Specifies to enforce the predefined configuration process during the sales transaction process. • `Skip`—Specifies to skip the configuration process during the quote creation process. • `System`—Specifies the system to determine whether the configuration process is required. The default value is `Skip`.	Optional	63.0
`configurationOptions`	Configuration Options Input	Configuration options during the ingestion process.	Optional	63.0

Context Input

Input representation of the context that's associated with a sales transaction for a quote or an order.

JSON example

```
{
  "contextDetails": {
    "contextId": "e055bb18-d4e8-41c3-881e-0132b9561708"
  }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`contextId`	String	ID of the context that represents the created session for the sales transaction. This property is supported only for a PATCH request.	Optional	63.0

Name	Type	Description	Required or Optional	Available Version
		If the `contextId` property isn't specified, the Place Sales Transaction API generates the context ID for the sales transaction.		

Context Node Input

Input representation of the details of the context nodes for ramp segments.

JSON example

```
"updatedNodes": [
   {
    "contextNodePath": [
     "4f23961a5c98806f89305e064c67b397e93f1bb8a2a7a3a80db506f1d4110ee9", // ContextId

      "0Q0xx0000004CPACA2", //Quote or OrderId
      "0QLxx0000004CfIGAU" // Quote Line ID or Order Line ID to update
    ],
    "contextNode": {
        "Discount": 10,
        "Quantity": 5
    }
  },
  {
    "contextNodePath": [
      "4f23961a5c98806f89305e064c67b397e93f1bb8a2a7a3a80db506f1d4110ee9",
      "0Q0xx0000004CPACA2",
      "2b6401d144904e10aa"
    ],
    "contextNode": {
        "Discount": 20,
        "Quantity": 15
    }
  }
]
```

Properties

Name	Type	Description	Required or Optional	Available Version
`contextNode`	Map<String, Object>	Details of the context node to be added, updated, or deleted.	Required	62.0
`contextNodePath`	String[]	Path to the context node to be added, updated, or deleted.	Required	62.0

Execution Settings Input

Input representation of the execution settings for a ramp deal.

JSON example

```
"executionSettings": {
     "executePricing": true,
     "executeConfigRules": false
 }
```

Properties

Name	Type	Description	Required or Optional	Available Version
`executeConfigRules`	Boolean	Indicates whether to run configuration rules (`true`) or not (`false`). The default value is `true`.	Optional	62.0
`executePricing`	Boolean	Indicates whether to run pricing request (`true`) or not (`false`). The default value is `true`.	Optional	62.0

Instant Pricing Input

Input representation to fetch the instant pricing details.

JSON example

```
{
  "correlationId": "1234567",
  "contextId": "",
  "records": [
    {
      "referenceId": "refQuote",
      "record": {
        "attributes": {
          "type": "Quote",
          "method": "POST"
        },
        "Name": "Test Quote Proration Pricing",
        "OpportunityId": "006xx000001a4ISAAY",
        "Pricebook2Id": "01sxx0000005ptpAAA"
      }
    },
    {
      "referenceId": "refQuoteLine",
      "record": {
        "attributes": {
          "type": "QuoteLineItem",
          "method": "POST"
        },
        "QuoteId": "refQuote",
        "PricebookEntryId": "01uxx0000008zHmAAI",
```

```
        "Product2Id": "01txx0000006jmWAAQ",
        "Quantity": 2,
        "UnitPrice": 25,
        "StartDate": "2022-09-28",
        "EndDate": "2028-09-27",
        "PeriodBoundary": "ANNIVERSARY",
        "BillingFrequency": "ANNUAL"
      }
    }
  ]
}
```

This example shows a sample request to specify grouping of lines based on criteria.

```
{
    "contextId": "",
    "correlationId": "",
    "records": [
        {
            "referenceId": "0Q0xx0000004DOSCA2",
            "record": {
                "attributes": {
                    "type": "Quote",
                    "method": "PUT"
                }
            }
        },
        {
            "referenceId": "0QLxx0000004F3gGAE",
            "record": {
                "attributes": {
                    "type": "QuoteLineItem",
                    "method": "PUT"
                },
                "Quantity": 5
            }
        }, {
            "referenceId": "0QLxx0000004F3hGAE",
            "record": {
                "attributes": {
                    "type": "QuoteLineItem",
                    "method": "PUT"
                },
                "Quantity": 2
            }
        },
        {
            "referenceId": "GroupId1",
            "record": {
                "attributes": {
                    "type": "QuoteLineGroup",
                    "method": "POST",
                    "action": "GroupBy",
                    "criteria": {
                        "Quantity": 5
```

```
                    }
                },
                "Name": "record"
            }
        }, {
            "referenceId": "GroupId2",
            "record": {
                "attributes": {
                    "type": "QuoteLineGroup",
                    "method": "POST",
                    "action": "GroupBy",
                    "criteria": {
                        "Quantity": 2
                    }
                },
                "Name": "record1",
            }
        }
    ]
}
```

This example shows a sample request for the initial grouping of the quote with the quote lines assigned to the first group.

```
{
    "contextId": "",
    "correlationId": "",
    "records": [
        {
            "referenceId": "0Q0xx0000004CAgCAM",
            "record": {
                "attributes": {
                    "type": "Quote",
                    "method": "PUT"
                }
            }
        },
        {
            "referenceId": "GroupId1",
            "record": {
                "attributes": {
                    "type": "QuoteLineGroup",
                    "method": "POST",
                    "action": "GroupAll"
                },
                "Name": "sample"
            }
        }
    ]
}
```

This example shows a sample request to ungroup a quote but retain the quote lines.

```
{
    "contextId": "",
    "correlationId": "",
```

```
    "records": [
        {
            "referenceId": "0Q0xx0000004CAgCAM",
            "record": {
                "attributes": {
                    "type": "Quote",
                    "method": "PUT"
                }
            }
        },
         {
            "referenceId": "0QLxx0000004CBYGA2",
            "record": {
                "attributes": {
                    "type": "QuoteLineItem",
                    "method": "PUT"
                },
                "Quantity": 2,
                "QuoteLineGroupId": null
            }
        },
        {
            "referenceId": "GroupId1",
            "record": {
                "attributes": {
                    "type": "QuoteLineGroup",
                    "method": "DELETE",
                    "action": "Ungroup"
                }
            }
        }
    ]
}
```

This example shows a sample request to create a new group.

```
{
    "contextId": "",
    "correlationId": "",
    "records": [
        {
            "referenceId": "0Q0xx0000004CAgCAM",
            "record": {
                "attributes": {
                    "type": "Quote",
                    "method": "PUT"
                }
            }
        },
        {
            "referenceId": "GroupId1",
            "record": {
                "attributes": {
                    "type": "QuoteLineGroup",
                    "method": "POST"
```

```
                },
                "Name": "sample"
            }
        }
    ]
}
```

This example shows a sample request to delete a group.

```
{
    "contextId": "",
    "correlationId": "",
    "records": [
        {
            "referenceId": "0Q0xx0000004CAgCAM",
            "record": {
                "attributes": {
                    "type": "Quote",
                    "method": "PUT"
                }
            }
        },
        {
            "referenceId": "GroupId1",
            "record": {
                "attributes": {
                    "type": "QuoteLineGroup",
                    "method": "DELETE",
                    "action": "DeleteGroup"
                },
                "Name": "sample"
            }
        }
    ]
}
```

This example shows a sample request to move a group.

```
{
    "contextId": "",
    "correlationId": "",
    "records": [
        {
            "referenceId": "0Q0xx0000004CAgCAM",
            "record": {
                "attributes": {
                    "type": "Quote",
                    "method": "PUT"
                }
            }
        },
        {
            "referenceId": "0QLxx0000004CBYGA2",
            "record": {
                "attributes": {
```

```
                    "type": "QuoteLineItem",
                    "method": "PUT"
                },
                "Quantity": 2
                "QuoteLineGroupId": "{@GroupId2}"
            }
        },
    ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`contextId`	String	The ID generated by the context service. If no context ID is provided, a new context is created.	Optional	59.0
`correlationId`	String	Client-generated ID for tracking multiple related API requests.	Optional	59.0
`records`	Object with Reference Input[]	List of pricing data to be fetched.	Required	59.0

Object Graph Input

Input representation of an sObject with a graph ID.

JSON example

```
{
  "graph": {
    "graphId": "1",
    "records": [
      {
        "referenceId": "refOrder",
        "record": {
          "attributes": {
            "type": "Order",
            "method": "POST"
          }
        }
      }
    ]
  }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`graphId`	String	The ID of the graph.	Required	60.0

Name	Type	Description	Required or Optional	Available Version
`records`	Object with Reference Input on page 726[]	List of the records to be ingested.	Required	60.0

Object Input Map

Input representation of an sObject record in a key-value map format.

JSON example

```
{
  "records": [
    {
      "referenceId": "refOrder",
      "record": {
        "attributes": {
          "type": "Order",
          "method": "PATCH",
          "id": "402xx000003KY5vJGH"
        },
        "Quantity": 5
      }
    }
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`attributes`	Map <String, String>	Configuration input for the record process. Valid values are: • `type`—Type of sales transaction such as Quote or Order. • `method`—HTTP methods such as POST, PATCH, and DELETE. • `id`—Unique identifier for the record. Required for PATCH and DELETE operations. • `criteria`—Criteria to group order or quote line items. For example, group order or quote line items	Required	60.0

Name	Type	Description	Required or Optional	Available Version
		based on a monthly billing frequency. • `action`—Action to group order or quote line items. Valid values are: – `GroupBy` – `Group` – `Ungroup` – `GroupAll` – `DeleteGroup`		

Object with Reference Input

Input representation of a list of records to be inserted or updated. To update a record, specify the record ID.

This is a sample request to create a sales transaction for an order line item.

JSON example

```
{
  "referenceId": "refOrderItem0",
  "record": {
    "attributes": {
      "type": "OrderItem",
      "method": "POST"
    },
    "OrderId": "@{refOrder.id}",
    "OrderActionId": "@{refOrderAction.id}",
    "PricebookEntryId": "01uRM000000igZG",
    "Quantity": 2
  }
}
```

This is a sample request to update an order line item.

JSON example

```
{
  "referenceId": "refOrderItem0",
  "record": {
    "attributes": {
      "type": "OrderItem",
      "method": "PATCH",
      "id": "402xx000003KY5vJGH"
    },
    "OrderId": "@{refOrder.id}",
    "OrderActionId": "@{refOrderAction.id}",
    "PricebookEntryId": "01uRM000000igZG",
    "Quantity": 2,
```

```
    "UnitPrice": 800
  }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`referenceId`	String	Reference ID that maps to the response and can be used as a reference in later subrecords.	Required	60.0
`records`	Object Input Map on page 725	Details of a record to be ingested.	Required	60.0

Place Order Input

Input representation of the request to create or update an order.

JSON example

```
{
  "pricingPref": "System",
  "configurationInput": "RunAndAllowErrors",
  "configurationOptions": {
    "validateProductCatalog": true,
    "validateAmendRenewCancel": true,
    "executeConfigurationRules": true,
    "addDefaultConfiguration": true
  },
  "graph": {
    "graphId": "graphId",
    "records": [
      {
        "referenceId": "refOrder",
        "record": {
          "attributes": {
            "type": "Order",
            "method": "POST",
            "Id": "POST"
          }
        }
      },
      {
         "referenceId":"refOrderItem",
         "record":{
         "attributes":{
             "type":"OrderItem",
             "method":"POST"
            },
            "OrderId":"@{refOrder.id}",
            "OrderActionId":"@{refOrderAction.id}",
            "ListPrice":"144.99",
```

```
            "Quantity":3,
            "PricebookEntryId":"01uxx0000008yXPAAY",
            "Product2Id":"01txx0000006i2UAAQ",
            "UnitPrice":"199.49"
        }
      }
    ]
  }
}
```

This example shows a sample request to define grouping of order items.

```
{
    "pricingPref": "system",
    "graph": {
        "graphId": "placeOrder",
        "records": [
            {
                "referenceId": "refOrder",
                "record": {
                    "attributes": {
                        "type": "Order",
                        "method": "PATCH",
                        "id": "801xx000003GZ9bAAG"
                    }
                }
            },
            {
                "referenceId": "refOlg1",
                "record": {
                    "attributes": {
                        "type": "OrderItemGroup",
                        "method": "POST"
                    },
                    "Name": "New Group",
                    "OrderId": "@{refOrder.id}"
                }
            }
        ]
    }
}
```

This example shows a sample request to ungroup order items.

```
{
    "pricingPref": "system",
    "graph": {
        "graphId": "placeOrder",
        "records": [
            {
                "referenceId": "refOrder",
                "record": {
                    "attributes": {
                        "type": "Order",
                        "method": "PATCH",
```

```
                        "id": "refOrder"
                    }
                }
            },
            {
                "referenceId": "refOlg1",
                "record": {
                    "attributes": {
                        "type": "OrderItemGroup",
                        "method": "DELETE",
                        "id": "refOlg1",
                        "action": "Ungroup"
                    }
                }
            }
        ]
    }
}
```

This example shows a sample request to create a new group.

```
{
    "pricingPref": "system",
    "graph": {
        "graphId": "placeOrder",
        "records": [
            {
                "referenceId": "refOrder",
                "record": {
                    "attributes": {
                        "type": "Order",
                        "method": "PATCH",
                        "id": "refOrder"
                    }
                }
            },
            {
                "referenceId": "refOlg",
                "record": {
                    "attributes": {
                        "type": "OrderItemGroup",
                        "method": "POST"
                    },
                    "Name": "New Group",
                    "OrderId": "@{refOrder.id}"
                }
            }
        ]
    }
}
```

This example shows a sample request to delete a group.

```
{
    "contextId": "",
```

```
    "correlationId": "",
    "records": [
        {
            "referenceId": "refOrder",
            "record": {
                "attributes": {
                    "type": "Order",
                    "method": "PATCH",
                    "id": "refOrder"
                }
            }
        },
        {
            "referenceId": "refOlg",
            "record": {
                "attributes": {
                    "type": "OrderItemGroup",
                    "method": "DELETE",
                    "id": "refOlg",
                    "action": "DeleteGroup"
                }
            }
        }
    ]
}
```

This example shows a sample request to group order items based on criteria.

```
{
    "pricingPref": "system",
    "graph": {
        "graphId": "placeOrder",
        "records": [
            {
                "referenceId": "refOrder",
                "record": {
                    "attributes": {
                        "type": "Order",
                        "method": "PATCH",
                        "id": "refOrder"
                    }
                }
            },
            {
                "referenceId": "g0",
                "record": {
                    "attributes": {
                        "type": "OrderItemGroup",
                        "method": "POST",
                        "action": "GroupBy",
                        "criteria": {
                            "BillingFrequency2": null
                        }
                    },
                    "Name": "Billing Frequency: ",
```

```
                    "OrderId": "@{refOrder.id}"
                }
            },
            {
                "referenceId": "g1",
                "record": {
                    "attributes": {
                        "type": "OrderItemGroup",
                        "method": "POST",
                        "action": "GroupBy",
                        "criteria": {
                            "BillingFrequency2": "Monthly"
                        }
                    },
                    "Name": "Billing Frequency: Monthly",
                    "OrderId": "@{refOrder.id}"
                }
            }
        ]
    }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`catalogRatesPref`	String	Rate card entries defined in the catalog that must be fetched for order items with usage-based pricing during the order creation process. Valid values are: • `Fetch`—Retrieves the rate card entries defined in the catalog for order items during the order creation process. • `Skip`—Skips the retrieval of rate card entries for order items during the order creation process. The default value is `Skip`. This property is available when the Usage-Based Selling feature is enabled.	Optional	62.0
`configurationInput`	String	Configuration input for the place order process. Valid values are: • `RunAndAllowErrors`—Specifies to run the configuration and to proceed order ingestion upon encountering any configuration errors. • `RunAndBlockErrors`—Specifies to run configuration and to block order	Optional	60.0

Name	Type	Description	Required or Optional	Available Version
		ingestion upon encountering any configuration errors. • `Skip`—Specifies to skip configuration. The default value is `RunAndBlockErrors`.		
`configurationOptions`	Configuration Options Input[]	Configuration options during the ingestion process.	Optional	60.0
`graph`	Object Graph Input	The sObject graph of the order payload to be ingested. You can perform create, update, or delete operations on objects from the Sales Transaction context definition by using this property. Additionally, perform create, update, or delete operations on custom objects and fields in your extended context definition.	Required	60.0
`pricingPref`	String	Pricing preference during the create order process. Valid values are: • `Force`—Specifies to force pricing during the order ingestion process. • `Skip`—Specifies to skip pricing during the order ingestion process. • `System`—Specifies the system to determine whether a pricing calculation is required. The default value is `System`.	Optional	60.0

Place Quote Input

Input representation of the request to create or update a quote.

JSON example

This example shows a sample request to create a quote.

```
{
  "pricingPref": "System",
  "configurationInput": "RunAndAllowErrors",
  "configurationOptions": {
    "validateProductCatalog": true,
    "validateAmendRenewCancel": true,
    "executeConfigurationRules": true,
    "addDefaultConfiguration": true
  },  "graph": {
    "graphId": "createQuote",
```

```
    "records": [{
        "referenceId": "refQuote",
        "record": {
          "attributes": {
            "type": "Quote",
            "method": "POST"
          },
          "opportunityId": "---",
          "quoteProp1": "value1",
          "quoteProp2": "value2"
        }
      },
      {
        "referenceId": "refQuoteLineItem1",
        "record": {
          "attributes": {
            "type": "QuoteLineItem",
            "method": "POST"
          },
          "QuoteLineItemProp1": "value1",
          "QuoteLineItemProp2": "value2"
        }
      },
      {
      "referenceId": "refQuoteLineItemAttribute",
      "record": {
        "attributes": {
              "type": "QuoteLineItemAttribute",
              "method": "POST"
       },
       "QuoteLineItemId": "@{refQuoteItem1.id}",
       "AttributeDefinitionId": "0tjxx0000000001AAA"
      }
}
    ]
  }
}
```

This example shows a sample request to insert, update, or delete a quote line item.

```
{
  "pricingPref": "System",
  "configurationInput": "skip",
  "graph": {
    "graphId": "updateQuote",
    "records": [
      {
        "referenceId": "refQuote",
        "record": {
          "attributes": {
            "type": "Quote",
            "method": "PATCH",
            "id": "0Q0xx0000004E2mCAE"
          },
          "Name": "Quote_Acme"
```

```
        }
      },
      {
        "referenceId": "refQuoteLineItemToCreate1",
        "record": {
          "attributes": {
            "type": "QuoteLineItem",
            "method": "POST"
          },
          "QuoteId": "0Q0xx0000004E2mCAE",
          "PricebookEntryId": "01uxx0000008yXPAAY",
          "Product2Id": "01txx0000006i2UAAQ",
          "Quantity": 2.0,
          "UnitPrice": 800.0,
          "PeriodBoundary": "Anniversary",
          "BillingFrequency": "Monthly",
          "StartDate": "2024-03-11"
        }
      },
      {
        "referenceId": "refQuoteLineItemToPatch2",
        "record": {
          "attributes": {
            "type": "QuoteLineItem",
            "method": "PATCH",
            "id": "0Q0xx0000004E2mCAE"
          },
          "Quantity": 2.0,
          "UnitPrice": 600.0
        }
      },
      {
        "referenceId": "refQuoteLineItemToDelete3",
        "record": {
          "attributes": {
            "type": "QuoteLineItem",
            "method": "DELETE",
            "id": "0Q0xx0000004E2mYLK"
          }
        }
      }
    ]
  }
}
```

This example shows a sample request to define grouping of quote line items.

```
{
   "pricingPref": "Force",
   "configurationInput": "skip",
   "graph": {
      "graphId": "groupLines",
      "records": [
         {
            "referenceId": "refQuote",
```

```
            "record": {
               "attributes": {
                  "type": "Quote",
                  "method": "PATCH",
                  "id":"0Q0xx0000004C99CAE"
               },
               "Name": "From Place Quote API"
            }
         },
         {
            "referenceId": "refQlg1",
            "record": {
               "attributes": {
                  "type": "QuoteLineGroup",
                  "method": "POST",
                  "action": "GroupBy",
                  "criteria": {
                    "Quantity": 1
                  }
               },
               "Name": "From Place Quote API Group",
               "QuoteId": "@{refQuote.id}"
            }
         },
         {
            "referenceId": "refQuoteItem1",
            "record": {
               "attributes": {
                  "type": "QuoteLineItem",
                  "method": "PATCH",
                  "id":"0QLxx0000004DJcGAM"
               },
               "QuoteLineGroupId": "@{refQlg1.id}",
               "Quantity": 1
            }
         }
      ]
   }
}
```

This example shows a sample request for the initial grouping of the quote with the quote lines assigned to the first group.

```
{
  "pricingPref": "Force",
  "graph": {
    "graphId": "test",
    "records": [
      {
        "referenceId": "refQuote",
        "record": {
          "attributes": {
            "type": "Quote",
            "method": "PATCH",
            "id": "0Q0xx0000004CAmCAM"
          }
```

```
        }
      },
      {
        "referenceId": "refQlg1",
        "record": {
          "attributes": {
            "type": "QuoteLineGroup",
            "method": "POST",
            "action": "GroupAll"
          },
          "Name": "From PQ API Group",
          "QuoteId": "@{refQuote.id}"
        }
      }
    ]
  }
}
```

This example shows a sample request to ungroup a quote but retain the quote lines.

```
{
   "pricingPref": "Force",
   "configurationInput": "skip",
   "graph": {
      "graphId": "test",
      "records": [
         {
            "referenceId": "refQuote",
            "record": {
               "attributes": {
                  "type": "Quote",
                  "method": "PATCH",
                  "id":"0Q0xx0000004C99CAE"
               },
               "Name": "From Place Quote API"
            }
         },
         {
            "referenceId": "refQlg1",
            "record": {
               "attributes": {
                  "type": "QuoteLineGroup",
                  "method": "DELETE",
                  "id": "{GroupId}",
                  "action": "Ungroup"
               }
            }
         }
      ]
   }
}
```

This example shows a sample request to create a new group.

```
{
   "pricingPref": "Force",
   "configurationInput": "skip",
   "graph": {
      "graphId": "test",
      "records": [
         {
            "referenceId": "refQuote",
            "record": {
               "attributes": {
                  "type": "Quote",
                  "method": "PATCH",
                  "id":"0Q0xx0000004C99CAE"
               },
               "Name": "From Place Quote API"
            }
         },
         {
            "referenceId": "refQlg1",
            "record": {
               "attributes": {
                  "type": "QuoteLineGroup",
                  "method": "POST"
               },
               "Name": "From PQ API Group",
               "QuoteId": "@{refQuote.id}"
            }
         }
      ]
   }
}
```

This example shows a sample request to delete a group.

```
{
   "pricingPref": "Force",
   "configurationInput": "skip",
   "graph": {
      "graphId": "test",
      "records": [
         {
            "referenceId": "refQuote",
            "record": {
               "attributes": {
                  "type": "Quote",
                  "method": "PATCH",
                  "id":"0Q0xx0000004C99CAE"
               },
               "Name": "From Place Quote API"
            }
         },
         {
            "referenceId": "refQlg1",
```

```
            "record": {
               "attributes": {
                  "type": "QuoteLineGroup",
                  "method": "DELETE",
                  "id": "{GroupId}",
                  "action": "DeleteGroup"
               }
            }
         }
      ]
   }
}
```

This example shows a sample request to move a group.

```
{
   "pricingPref": "Force",
   "configurationInput": "skip",
   "graph": {
      "graphId": "test",
      "records": [
         {
            "referenceId": "refQuote",
            "record": {
               "attributes": {
                  "type": "Quote",
                  "method": "PATCH",
                  "id":"0Q0xx0000004C99CAE"
               },
               "Name": "From PlaceQuote Api"
            }
         },
        {
            "referenceId": "0QLxx0000004CBYGA2",
            "record": {
                "attributes": {
                    "type": "QuoteLineItem",
                    "method": "PATCH"
                    "id": "0QLxx0000004CBYGA2"
                },
                "Quantity": 2,
                "QuoteLineGroupId": "@{GroupId2}"
            }
      ]
   }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`catalogRatesPref`	String	Rate card entries defined in the catalog that must be fetched for quote line items	Optional	62.0

Name	Type	Description	Required or Optional	Available Version
		with usage-based pricing during the quote creation process. Valid values are: • `Fetch`—Retrieves the rate card entries defined in the catalog for quote line items during the quote creation process. • `Skip`—Skips the retrieval of rate card entries for quote line items during the quote creation process. The default value is `Skip`. This property is available when the Usage-Based Selling feature is enabled.		
`configurationInput`	String	Configuration input for the place quote process. Valid values are: • `RunAndAllowErrors` • `RunAndBlockErrors` • `Skip` The default value is `RunAndBlockErrors`.	Optional	60.0
`configurationOptions`	Configuration Options Input	Configuration options during the ingestion process.	Optional	60.0
`graph`	Object Graph Input	The sObject graph representing the quote structure. You can perform create, update, or delete operations on objects from the Sales Transaction context definition by using this property. Additionally, perform create, update, or delete operations on custom objects and fields in your extended context definition.	Required	60.0
`pricingPref`	String	Pricing preference during the quote process. Valid values are: • `Force` • `Skip` • `System` The default value is `System`.	Optional	60.0

Place Sales Transaction Input

Input representation of the details of the request to place a sales transaction, such as a quote or an order.

JSON example

This is a sample request to create a sales transaction for a quote.

```
{
  "pricingPref": "system",
  "catalogRatesPref" "skip",
  "configurationPref": {
    "configurationMethod": "Skip",
    "configurationOptions": {
      "validateProductCatalog": true,
      "validateAmendRenewCancel": true,
      "executeConfigurationRules": true,
      "addDefaultConfiguration": true
    }
  },
  "contextDetails": {
    "contextId": "e055bb18-d4e8-41c3-881e-0132b9561708"
  },
  "graph": {
    "graphId": "createQuote",
    "records": [
      {
        "referenceId": "refQuote",
        "record": {
          "attributes": {
            "method": "POST",
            "type": "Quote"
          },
          "Name": "Quote_Acme",
          "Pricebook2Id": "01sDU000000JvhbYAC"
        }
      },
      {
        "referenceId": "refQuoteLine0",
        "record": {
          "attributes": {
            "type": "QuoteLineItem",
            "method": "POST"
          },
          "QuoteId": "@{refQuote.id}",
          "Product2Id": "01tDU000000F7b8YAC",
          "PricebookEntryId": "01uDU000000fxt2YAA",
          "UnitPrice": 100,
          "Quantity": "1",
          "StartDate": "2024-10-29",
          "EndDate": "2025-03-01",
          "PeriodBoundary": "Anniversary"
        }
      }
    ]
  }
}
```

This is a sample request to insert, update, or delete a quote line item.

```
{
  {
  "pricingPref": "system",
  "catalogRatesPref": "skip",
  "configurationPref": {
    "configurationMethod": "Skip",
    "configurationOptions": {
      "validateProductCatalog": true,
      "validateAmendRenewCancel": true,
      "executeConfigurationRules": true,
      "addDefaultConfiguration": true
    }
  },
  "contextDetails": {
    "contextId": "e055bb18-d4e8-41c3-881e-0132b9561708"
  },
  "graph": {
    "graphId": "updateQuote",
    "records": [
      {
        "referenceId": "refQuote",
        "record": {
          "attributes": {
            "method": "PATCH",
            "type": "Quote",
            "id": "801xx000003GZ9bAAG"
          }
        }
      },
      {
        "referenceId": "refQuoteLine0",
        "record": {
          "attributes": {
            "type": "QuoteLineItem",
            "method": "PATCH",
            "id": "402xx000003KY5vJGH"
          },
          "Quantity": "5"
        }
      }
    ]
  }
}
```

This is a sample request to define grouping of quote line items.

```
{
  "pricingPref": "Force",
  "catalogRatesPref" "skip",
  "configurationPref": {
    "configurationMethod": "Skip",
    "configurationOptions": {
      "validateProductCatalog": true,
```

```
      "validateAmendRenewCancel": true,
      "executeConfigurationRules": true,
      "addDefaultConfiguration": true
    }
  },
  "contextDetails": {
    "contextId": "e055bb18-d4e8-41c3-881e-0132b9561708"
  },
  "graph": {
    "graphId": "groupQuoteLines",
    "records": [
      {
        "referenceId": "refQuote",
        "record": {
          "attributes": {
            "method": "PATCH",
            "type": "Quote",
            "id": "801xx000003GZ9bAAG"
          }
        }
      },
      {
        "referenceId": "refQuoteLine0",
        "record": {
          "attributes": {
            "type": "QuoteLineItem",
            "method": "PATCH",
            "id": "402xx000003KY5vJGH"
          },
          "QuoteLineGroupId": "@{refQuote.id}"
        }
      }
    ]
  }
}
```

This is s a sample request for the initial grouping of the quote with all the quote lines assigned to the first group.

```
{
  "pricingPref": "Force",
  "catalogRatesPref" "skip",
  "graph": {
    "graphId": "groupQuote",
    "records": [
      {
        "referenceId": "refQuote",
        "record": {
          "attributes": {
            "type": "Quote",
            "method": "PATCH",
            "id": "0Q0xx0000004CAmCAM"
          }
        }
      },
      {
```

```
        "referenceId": "refQlg1",
        "record": {
          "attributes": {
            "type": "QuoteLineGroup",
            "method": "POST",
            "action": "GroupAll"
          },
          "Name": "From PST API Group",
          "QuoteId": "@{refQuote.id}"
        }
      }
    ]
  }
}
```

This is a sample request to ungroup a quote but retain the quote lines.

```
{
   "catalogRatesPref" "skip",
   "pricingPref": "Force",
   "graph": {
      "graphId": "ungroupQuote",
      "records": [
         {
            "referenceId": "refQuote",
            "record": {
               "attributes": {
                  "type": "Quote",
                  "method": "PATCH",
                  "id":"0Q0xx0000004C99CAE"
               },
               "Name": "Grouped Quote with PST API"
            }
         },
         {
            "referenceId": "refQlg1",
            "record": {
               "attributes": {
                  "type": "QuoteLineGroup",
                  "method": "DELETE",
                  "id": "402xx000003KY5vJGH",
                  "action": "Ungroup"
               }
            }
         }
      ]
   }
}
```

This is a sample request to create a new group.

```
{
   "catalogRatesPref" "skip",
   "pricingPref": "Force",
   "graph": {
```

```
      "graphId": "createGroup",
      "records": [
         {
            "referenceId": "refQuote",
            "record": {
               "attributes": {
                  "type": "Quote",
                  "method": "PATCH",
                  "id":"0Q0xx0000004C99CAE"
               },
               "Name": "Grouped Quote with PST API"
            }
         },
         {
            "referenceId": "refQlg1",
            "record": {
               "attributes": {
                  "type": "QuoteLineGroup",
                  "method": "POST"
               },
               "Name": "From PQ API Group",
               "QuoteId": "@{refQuote.id}"
            }
         }
      ]
   }
}
```

This example shows a sample request to delete a group.

```
{
   "catalogRatesPref" "skip",
   "pricingPref": "Force",
   "graph": {
      "graphId": "deleteGroup",
      "records": [
         {
            "referenceId": "refQuote",
            "record": {
               "attributes": {
                  "type": "Quote",
                  "method": "PATCH",
                  "id":"0Q0xx0000004C99CAE"
               },
               "Name": "Grouped Quote with PST API"
            }
         },
         {
            "referenceId": "refQlg1",
            "record": {
               "attributes": {
                  "type": "QuoteLineGroup",
                  "method": "DELETE",
                  "id": "402xx000003KY5vJGH",
                  "action": "DeleteGroup"
```

```
                }
            }
        }
    ]
  }
}
```

This is a sample request to group order items based on criteria.

```
{
    "catalogRatesPref" "skip",
    "pricingPref": "Force",
    "graph": {
        "graphId": "groupOrderItems",
        "records": [
            {
                "referenceId": "refOrder",
                "record": {
                    "attributes": {
                        "type": "Order",
                        "method": "PATCH",
                        "id": "0Q0xx0000004C99CAE"
                    }
                }
            },
            {
                "referenceId": "refOlg1",
                "record": {
                    "attributes": {
                        "type": "OrderItemGroup",
                        "method": "POST",
                        "action": "GroupBy",
                        "criteria": {
                            "BillingFrequency2": null
                        }
                    },
                    "Name": "Billing Frequency: ",
                    "OrderId": "@{refOrder.id}"
                }
            },
            {
                "referenceId": "g1",
                "record": {
                    "attributes": {
                        "type": "OrderItemGroup",
                        "method": "POST",
                        "action": "GroupBy",
                        "criteria": {
                            "BillingFrequency2": "Monthly"
                        }
                    },
                    "Name": "Billing Frequency: Monthly",
                    "OrderId": "@{refOrder.id}"
                }
            }
```

```
        ]
    }
}
```

This is a sample request to save changes to a ramp deal by using context ID. The context ID is returned by the Ramp Deal APIs. See Create Ramp Deal (POST).

```
{
  "pricingPref": "force",
  "contextDetails": {
    "contextId": "f1c9e3e1c335f7959a88de09d3a867cc2b95e08709b99de8e2edeb8f5039e8ed",
    "scope": "Session"
  },
  "graph": {
    "graphId": "updateQuote",
    "records": [
      {
        "referenceId": "refQuote",
        "record": {
          "attributes": {
            "type": "Quote",
            "method": "PATCH",
            "id": "0Q0xx0000004DQ4CAM"
          }
        }
      }
    ]
  }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`catalogRatesPref`	String	Rate card entries defined in the catalog that must be fetched for sales items with usage-based pricing during the creation of the sales transaction. Valid values are: • `Fetch`—Retrieves the rate card entries defined in the catalog for sales items during the creation of the sales transaction. • `Skip`—Skips the retrieval of rate card entries for sales items during the creation of the sales transaction. The default value is `Skip`. This property is available when the Usage-Based Selling feature is enabled.	Optional	63.0

Name	Type	Description	Required or Optional	Available Version
`configurationPref`	Configurator Preference Input	Configuration preference during the quote process. These preferences ensure that quotes are defined as per the requirement.	Optional	63.0
`contextDetails`	Context Input	Context details that are created for a sales transaction.	Required if the `graph` property isn't specified.	63.0
`graph`	Object Graph Input	The sObject graph of the sales transaction to be ingested. You can perform create, update, or delete operations on objects from the Sales Transaction context definition by using this property. Additionally, perform create, update, or delete operations on custom objects and fields in your extended context definition.	Required if the `contextDetails` property isn't specified.	63.0
`pricingPref`	String	Pricing preference during the creation of a sales transaction. Valid values are: • `Force`—Specifies to enforce pricing during the creation of sales transactions. • `Skip`—Specifies to skip pricing during the creation of sales transactions. • `System`—Specifies the system to determine whether a pricing calculation is required. The default value is `System`.	Optional	63.0

Create Ramp Deal Input

Input representation of the request to create a ramp deal.

JSON example

```
{
  "transactionId": "0Q0xx0000004C92CAE",
  "transactionLineId": "0QLxx0000004C9VGAU",
  "subscriptionTerm": 14,
  "subscriptionTermUnit": "MONTHS",
  "trialTerm": 45,
  "trialTermUnit": "DAYS",
  "segmentType": "YEARLY",
  "executionSettings": {
       "executePricing": true,
       "executeConfigRules": false
```

```
    }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`execution Settings`	Execution Settings Input[]	Settings to run the pricing or configuration rules.	Optional	62.0
`segmentType`	String	Type of segment that the user wants to create. Valid values are: • `FREE_TRIAL` • `CUSTOM` • `YEARLY`	Required	62.0
`subscription Term`	Integer	Subscription length of the term-defined product.	Required	62.0
`subscription TermUnit`	String	Unit of time for the subscription length. Valid value is: • `MONTHS`	Required	62.0
`transactionId`	String	ID of the sales transaction that's configured, such as quote or order.	Required	62.0
`transaction LineId`	String	Quote line item ID or order item ID that the price ramp is created for.	Required	62.0
`trialTerm`	Integer	Length of the trial period, if any.	Optional	62.0
`trialTermUnit`	String	Unit of time for the trial period. Valid value is: • `DAYS`	Optional. Required if `trialTerm` property is specified.	62.0

Delete Ramp Deal Input

Input representation of the request to delete a ramp deal.

JSON example

```
{
  "rampDealIds": [
    "0Q0xx0000004CDxCAM",
    "0QLxx0000004CSOGA2"
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`rampDealIds`	String[]	Ramp identifier on the quote line item or order item.	Required	62.0

Update Ramp Deal Input

Input representation of the request to update a ramp deal.

JSON example

```
{
  "executionSettings": {
       "executePricing": true,
       "executeConfigRules": false
  },
  "addedNodes": [
    {
      "contextNodePath": [
        "4f23961a5c98806f89305e064c67b397e93f1bb8a2a7a3a80db506f1d4110ee9", // Context
 ID
        "0Q0xx0000004CPACA2", //Quote or Order ID
        "RandomUUID" // random UUID for Quote Line Item or Order Item ID
      ],
      "contextNode": {
         "Discount": 10,
         "Quantity": 5,
         "ItemSegmentName": "Year 5",
         "StartDate":"2024-09-07T00:00:00.000Z",
         "EndDate":"2024-09-07T00:00:00.000Z"
      }
    }
  ],
  "updatedNodes": [
     {
      "contextNodePath": [
        "4f23961a5c98806f89305e064c67b397e93f1bb8a2a7a3a80db506f1d4110ee9", // Context
 ID
        "0Q0xx0000004CPACA2", //Quote or Order ID
        "0QLxx0000004CfIGAU" // Quote Line ID or Order Line ID to update
      ],
      "contextNode": {
          "Discount": 10,
          "Quantity": 5
      }
    }
  ],
  "deletedNodes": [
    {
      "contextNodePath": [
        "4f23961a5c98806f89305e064c67b397e93f1bb8a2a7a3a80db506f1d4110ee9",
```

```
        "0Q0xx0000004CPACA2",
        "0QLxx0000004CfIGAU" // Quote Line Item ID to delete
      ]
    }
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`addedNodes`	Context Node Input[]	Details of the nodes to be added.	Required	62.0
`deletedNodes`	Context Node Input[]	Details of the nodes to be deleted.	Required	62.0
`execution Settings`	Execution Settings Input[]	Settings to run the pricing or configuration rules.	Optional	62.0
`updatedNodes`	Context Node Input[]	Details of the nodes to be updated.	Required	62.0

Response Bodies

Learn more about the available response bodies.

Instant Pricing
Output representation containing the results of the instant pricing request.

Object Reference
Output representation of an sObject with a reference ID along with any potential error.

Place Order Error Response
Output representation of the error response for the place order request.

Place Order Response
Output representation of the request to create or update an order.

Place Quote Error Response
Output representation of the error responses of a place quote request.

Place Quote
Output representation of the request to create or update a quote.

Place Sales Transaction
Output representation of the request to create a sales transaction.

Place Sales Transaction Context
Output representation of the context details that are associated with a sales transaction.

Sales Transaction Error Response
Output representation of the error details associated with the Place Sales Transaction API.

Ramp Deal Service Error Response

Output representation of the details of errors encountered during the processing of the API request.

Ramp Deal Service

Output representation of the details of a created, updated, or deleted ramp deal.

Instant Pricing

Output representation containing the results of the instant pricing request.

Sample Response

```
{
  "correlationid": "123",
  "contextid": "abcl23",
  "records": [
    {
      "referenceid": "refQuote",
      "record": {
        "attributes": {
          "type": "Quote",
          "method": "POST"
        },
        "quantity": "2"
      },
      "error": {
        "errorCode": "INVALID_API_INPUT",
        "message": "Reference Id format is irrelevant."
      }
    }
  ],
  "success": true
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`contextId`	String	Context ID returned by the context service.	Small, 59.0	59.0
`correlationId`	String	Client-generated ID for tracking multiple related API calls.	Small, 59.0	59.0
`records`	Object Reference[]	List of records related to pricing results.	Small, 59.0	59.0
`success`	Boolean	Indicates whether the fetching of instant pricing is successful (`true`) or not (`false`).	Small, 59.0	59.0

Object Reference

Output representation of an sObject with a reference ID along with any potential error.

Sample Response

```
{
  "referenceid": "refQuote",
  "record": {
    "attributes": {
      "type": "Quote",
      "method": "POST"
    },
    "quantity": "2"
  },
  "error": {
    "errorCode": "INVALID_API_INPUT",
    "message": "Reference Id format is irrelevant."
  }
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
referenceId	String	ID that identifies the specific Salesforce object that's returned in the API response.	Small, 59.0	59.0
record	Map<String, Object>	The sObject record data represented as a map of attribute names to their values.	Small, 59.0	59.0
error	[illegible]	Detailed information about any error associated with the sObject in the response.	Small, 59.0	59.0

Place Order Error Response

Output representation of the error response for the place order request.

Property Name	Type	Description	Filter Group and Version	Available Version
errorCode	String	Code representing the type of error encountered during the place order create request.	Small, 60.0	60.0
message	String	Message stating the reason for the error, if any.	Small, 60.0	60.0
referenceId	String	Reference ID associated with the specific error instance for tracking and reference purposes.	Small, 60.0	60.0

Place Order Response

Output representation of the request to create or update an order.

JSON example

```
{
  "requestId": "16PRM0000004DBq",
  "orderId": "801S70000001VKgIAM",
  "success": true,
  "errors": [],
  "statusURL": "/services/data/vXX.X/sobjects/AsyncOperationTracker/16PRM0000004DBq"
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`errors`	Place OrderError Response on page 752[]	List of errors encountered during the synchronous processing.	Small, 60.0	60.0
`orderId`	String	ID of the order created after a successful request.	Small, 60.0	60.0
`requestId`	String	Request ID of the process to query asynchronous status of the place order API.	Small, 60.0	60.0
`statusURL`	String	Asynchronous status URL of the request, if available.	Small, 60.0	60.0
`success`	Boolean	Indicates whether the synchronous part of the processing is successful (`true`) or not (`false`).	Small, 60.0	60.0

Place Quote Error Response

Output representation of the error responses of a place quote request.

JSON Example

```
{
 "errorCode": "INVALID_API_INPUT",
 "message": "Include record type and method in the request and try again.",
 "referenceId": "refQuoteItem2"
 }
```

Property Name	Type	Description	Filter Group and Version	Available Version
`errorCode`	String	Error code representing the type of error encountered in the create place quote request.	Small, 60.0	60.0
`message`	String	Message stating the reason for the error, if any.	Small, 60.0	60.0

Property Name	Type	Description	Filter Group and Version	Available Version
referenceId	String	Reference ID associated with the specific error instance for tracking and reference purposes.	Small, 60.0	60.0

Place Quote

Output representation of the request to create or update a quote.

JSON Example

This example shows a sample response of the place quote request.

```
{
    "quoteId": "0Q0xx0000004E2mCAE",
    "requestIdentifier": "95Txx0000004Cx2",
    "responseError": [],
   "statusURL": "/services/data/v60.0/sobjects/RevenueAsyncOperation/95Txx0000004Cx2EAE",

    "success": true
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
quoteId	String	ID of the quote created after a successful request.	Small, 60.0	60.0
request Identifier	String	Unique request identifier that can be used to poll the async request.	Big, 60.0	60.0
responseError	Place Quote Error Response []	List of errors encountered during the synchronous processing.	Small, 60.0	60.0
statusURL	String	Asynchronous status URL to track the operation, if available.	Big, 60.0	60.0
success	Boolean	Indicates whether the synchronous part of the processing is successful (true) or not (false).	Small, 60.0	60.0

Place Sales Transaction

Output representation of the request to create a sales transaction.

JSON example

```
{
  "contextDetails": {
    "contextId": "e055bb18-d4e8-41c3-881e-0132b9561708",
    "isBuiltInTransaction": true
  },
  "errorResponse": {
```

```
    "errorCode": "INVALID_API_INPUT",
    "message": "Include record type and method in the request and try again.",
    "referenceId": "refQuoteItem2"
  },
  "isSuccess": true,
  "salesTransactionId": "0Q0xx0000004CNYCA2",
  "statusUrl": null,
  "trackerId": null
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
contextDetails	Sales Transaction Context	Details of the context that's created for the sales transaction.	Small, 63.0	63.0
errorResponse	Sales Transaction Error Response[]	Details of the error if the operation fails.	Small, 63.0	63.0
isSuccess	Boolean	Indicates if the operation is successful (true) or not (false).	Small, 63.0	63.0
salesTransactionId	String	ID of the sales transaction, such as a quote or an order.	Small, 63.0	63.0
statusUrl	String	URL to check the status of the operation.	Small, 63.0	63.0
trackerId	String	Unique identifier assigned to a specific operation or request that's used for tracking and referencing the operation.	Small, 63.0	63.0

Place Sales Transaction Context

Output representation of the context details that are associated with a sales transaction.

JSON example

```
{
  "contextDetails": {
    "contextId": "e055bb18-d4e8-41c3-881e-0132b9561708",
    "isBuiltInTransaction": true
  }
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
contextId	String	ID of the context that's created for a session of the sales transaction.	Small, 63.0	63.0

Property Name	Type	Description	Filter Group and Version	Available Version
`isBuiltInTransaction`	Boolean	Indicates whether a new context ID is created for the sales transaction (`true`) or not (`false`). If the `contextId` property isn't specified, the Place Sales Transaction API generates it.	Small, 63.0	63.0

Sales Transaction Error Response

Output representation of the error details associated with the Place Sales Transaction API.

JSON example

```
{
  "errorResponse": {
    "errorCode": "INVALID_API_INPUT",
    "message": "Include record type and method in the request and try again.",
    "referenceId": "refQuoteItem2"
  }
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`errorCode`	String	Code for the resultant error.	Small, 63.0	63.0
`message`	String	Error message for the resultant error.	Small, 63.0	63.0
`referenceId`	String	Unique ID that's associated with the specific error for tracking and reference purposes.	Small, 63.0	63.0

Ramp Deal Service Error Response

Output representation of the details of errors encountered during the processing of the API request.

Property Name	Type	Description	Filter Group and Version	Available Version
`errorCode`	String	Error code from the API request. For example, `INVALID_INPUT_ERROR`.	Small, 62.0	62.0
`message`	String	Error message from the API request.	Small, 62.0	62.0

Ramp Deal Service

Output representation of the details of a created, updated, or deleted ramp deal.

JSON Example

This example shows the sample response for the create, update, or view ramp deal requests.


```
{
  "correlationId": "0QLDU0000002t0Z4AQ",
  "errors": [],
  "salesTransactionContext": {
    "SalesTransaction": [
      {
        "LegalEntity": null,
        "Account": null,
        "HeaderDistributionType": null,
        "BillingCity": null,
        "AccountBusinessType": null,
        "HeaderDiscountType": null,
        "businessObjectType": "Quote",
        "QuoteAccount": null,
        "EmployeeCount": null,
        "SalesTransactionName": "WarrantyPriceRampAR",
        "StartDate": null,
        "HeaderDiscountValue": null,
        "SalesTransactionType": null,
        "Pricebook": "01sDU000000JGf4YAG",
        "Opportunity": null,
        "ShippingCountry": null,
        "ShippingCity": null,
        "BillingPostalCode": null,
        "id": "0Q0DU0000002f3d0AA",
        "BillToContact": null,
        "CalculationStatus": "CompletedWithTax",
        "Status": "Draft",
        "LastPricedDate": null,
        "Subtotal": 108.47145205479453,
        "OriginalActionType": null,
        "TotalAmount": 99.98,
        "CurrencyIsoCode": null,
        "ShippingStreet": null,
        "SalesTransactionItem": [
          {
            "LegalEntity": null,
            "ProductName": "Warranty",
            "businessObjectType": "QuoteLineItem",
            "Product": "01tDU000000EsWSYA0",
            "ItemIsPrimarySegment": true,
            "ListPrice": 49.99,
            "ValidationResult": null,
            "StartDate": "2024-08-23T00:00:00.000Z",
            "ContractVolumePasId": null,
            "BillingTreatment": null,
            "PeriodBoundaryStartMonth": null,
            "SalesTransactionSourceAsset": null,
            "id": "0QLDU0000002t0Z4AQ",
            "PartnerDiscountPercent": null,
            "PriceWaterFall": "0QLDU0000002t0Z4AQ:548201414593252",
```

```
            "ItemProductRecipient": null,
            "BillingFrequency": "Annual",
            "ProductCode": "W001",
            "DerivedPricingAttribute": null,
            "TaxTreatment": "1ttDU0000001oGKYAY",
            "Subtotal": 8.491452054794523,
            "ItemRampIdentifier": "RDI5b5ce52b2db4484",
            "ItemSegmentName": "Trial",
            "PricebookEntry": "01uDU000000f4LSYAY",
            "DiscountAmount": null,
            "PricingTermCount": 0.0849315068493151,
            "NetUnitPrice": 0,
            "ItemEffectiveGrantDate": null,
            "ProductCategory": null,
            "SalesTransactionAction": null,
            "SalesTransactionActionType": null,
            "SalesTransactionItemGroup": null,
            "PeriodBoundaryDay": null,
            "LineItemDistributionType": null,
            "ProrationPolicy": "0muDU00000029ryYAA",
            "ContractDiscountType": null,
            "TransactionType": null,
            "ParentReference": null,
            "Discount": 100,
            "ProductSellingModel": "0jPDU00000029zb2AA",
            "PricingTermUnit": "Annual",
            "PricingSource": null,
            "StockKeepingUnit": null,
            "PartnerUnitPrice": null,
            "ItemTotalAdjustmentAmount": -8.491452054794523,
            "SalesTransactionItemSource": "0QLDU0000002t0Z4AQ",
            "ContractAttributePasId": null,
            "SubscriptionTerm": 1,
            "SellingModelType": "TermDefined",
            "EndQuantity": 2,
            "NetTotalPrice": 0,
            "TotalLineAmount": 8.491452054794523,
            "ItemSegmentType": "FreeTrial",
            "ProductBasedOn": null,
            "Deleted": null,
            "BillingReference": null,
            "ArePartialPeriodsAllowed": true,
            "ItemRecordedPrice": null,
            "CustomProductName": "Warranty",
            "ItemSegmentIdentifier": "SEG4380006a1c2b416",
            "SalesTransactionItemParent": "0Q0DU0000002f3d0AA",
            "Quantity": 2,
            "PeriodBoundary": "Anniversary",
            "ContractDiscountValue": null,
            "LineItemDiscountValue": null,
            "ContractId": null,
            "EndDate": "2024-09-22T00:00:00.000Z",
            "ItemGroupSummarySubtotal": null,
            "IsContracted": null,
```

```
    "UnitPrice": 49.99,
    "StartQuantity": 0,
    "ContractPrice": null,
    "TotalPrice": 0,
    "ItemPath": null,
    "LineItemDiscountType": null
  },
  {
    "LegalEntity": null,
    "ProductName": "Warranty",
    "businessObjectType": "QuoteLineItem",
    "Product": "01tDU000000EsWSYA0",
    "ItemIsPrimarySegment": false,
    "ListPrice": 49.99,
    "ValidationResult": null,
    "StartDate": "2024-09-23T00:00:00.000Z",
    "ContractVolumePasId": null,
    "BillingTreatment": null,
    "PeriodBoundaryStartMonth": null,
    "SalesTransactionSourceAsset": null,
    "id": "0QLDU0000003CZ94AM",
    "PartnerDiscountPercent": null,
    "PriceWaterFall": "0QLDU0000003CZ94AM:548201414593252",
    "ItemProductRecipient": null,
    "BillingFrequency": "Annual",
    "ProductCode": "W001",
    "DerivedPricingAttribute": null,
    "TaxTreatment": "1ttDU0000001oGKYAY",
    "Subtotal": 99.98,
    "ItemRampIdentifier": "RDI5b5ce52b2db4484",
    "ItemSegmentName": "Year-1",
    "PricebookEntry": "01uDU000000f4LSYAY",
    "DiscountAmount": null,
    "PricingTermCount": 1,
    "NetUnitPrice": 49.99,
    "ItemEffectiveGrantDate": null,
    "ProductCategory": null,
    "SalesTransactionAction": null,
    "SalesTransactionActionType": null,
    "SalesTransactionItemGroup": null,
    "PeriodBoundaryDay": null,
    "LineItemDistributionType": null,
    "ProrationPolicy": "0muDU00000029ryYAA",
    "ContractDiscountType": null,
    "TransactionType": null,
    "ParentReference": null,
    "Discount": null,
    "ProductSellingModel": "0jPDU00000029zb2AA",
    "PricingTermUnit": "Annual",
    "PricingSource": null,
    "StockKeepingUnit": null,
    "PartnerUnitPrice": null,
    "ItemTotalAdjustmentAmount": 0,
    "SalesTransactionItemSource": "0QLDU0000003CZ94AM",
```

```
            "ContractAttributePasId": null,
            "SubscriptionTerm": 1,
            "SellingModelType": "TermDefined",
            "EndQuantity": 2,
            "NetTotalPrice": 99.98,
            "TotalLineAmount": 99.98,
            "ItemSegmentType": "Yearly",
            "ProductBasedOn": null,
            "Deleted": null,
            "BillingReference": null,
            "ArePartialPeriodsAllowed": true,
            "ItemRecordedPrice": null,
            "CustomProductName": "Warranty",
            "ItemSegmentIdentifier": "SEG73ad7378e1ed4c5",
            "SalesTransactionItemParent": "0Q0DU0000002f3d0AA",
            "Quantity": 2,
            "PeriodBoundary": "Anniversary",
            "ContractDiscountValue": null,
            "LineItemDiscountValue": null,
            "ContractId": null,
            "EndDate": "2025-08-22T00:00:00.000Z",
            "ItemGroupSummarySubtotal": null,
            "IsContracted": null,
            "UnitPrice": 49.99,
            "StartQuantity": 0,
            "ContractPrice": null,
            "TotalPrice": 99.98,
            "ItemPath": null,
            "LineItemDiscountType": null
          }
        ],
        "AppUsageAssignment": [
          {
            "businessObjectType": "AppUsageAssignment",
            "ParentReference": null,
            "Record": "0Q0DU0000002f3d0AA",
            "id": "0j8DU0000002VKiYAM",
            "AppUsageType": "RevenueLifecycleManagement"
          }
        ],
        "BillingCountry": null,
        "BillingStreet": null,
        "ShippingPostalCode": null,
        "SalesTransactionSource": "0Q0DU0000002f3d0AA",
        "PrimaryIndustry": null,
        "ShippingState": null,
        "HeaderDistributionLogic": null,
        "Contract": null,
        "BillingState": null,
        "AnnualRevenue": null,
        "EffectiveDate": null
      }
    ]
  },
```

```
  "success": true,
  "transactionContextId":
"d3fd83b007418ce4980340313b40fd45665b194973486ebac3674c2b8002336f"
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`correlationId`	String	Resource ID to correlate the API request with the response.	Small, 62.0	62.0
`errors`	Ramp Deal Service Error Response[]	List of errors encountered during the processing of the API request.	Small, 62.0	62.0
`salesTransactionContext`	Map<String, Object>	Context object for the sales transaction with updated segment details.	Small, 62.0	62.0
`success`	Boolean	Indicates whether the API request is successful (`true`) or not (`false`).	Small, 62.0	62.0
`transactionContextId`	String	ID of the sales transaction context record instance.	Small, 62.0	62.0

Quote and Order Capture Apex Reference

Use built-in Apex classes and interfaces grouped by namespace.

CommerceOrders Namespace

The `CommerceOrders` namespace provides classes and methods to place orders with integrated pricing, configuration, and validation.

CommerceTax Namespace

Manage the communication between Salesforce and an external tax engine.

PlaceQuote Namespace

The `PlaceQuote` namespace provides classes and methods to create or update quotes with pricing preferences and configuration options.

RevSalesTrxn Namespace

Create a sales transaction, such as a quote or an order, with integrated pricing and configuration. Additionally, update an order or a quote, and insert and delete order or quote line items to calculate the estimated tax.

CommerceOrders Namespace

The `CommerceOrders` namespace provides classes and methods to place orders with integrated pricing, configuration, and validation.

The `CommerceOrders` namespace includes these classes.

CatalogRatesPreferenceEnum Enum

Specifies the rate card entries defined in the catalog that must be fetched for order items, with usage-based selling during the order creation process.

ConfigurationInputEnum Enum

Specifies the configuration input for the request to place an order.

ConfigurationOptionsInput Class

Contains methods and properties to set the configuration options for the input to the product configurator.

GraphRequest Class

Contains constructors and properties to set the graph ID and a list of records to be ingested. The list of records is specified in a key-value map format that contains the field values of an order.

PlaceOrderExecutor Class

Contains methods to place an order with details of the graph request, pricing preferences, and configuration options.

PlaceOrderResult Class

Contains properties to hold the response to the place order request.

PricingPreferenceEnum Enum

Specifies the pricing preference during the create order process.

RecordResource Class

Contains constructors and properties to create a record object from field values of an order.

RecordWithReferenceRequest Class

Contains constructors and properties to associate a record object with a reference identifier.

CatalogRatesPreferenceEnum Enum

Specifies the rate card entries defined in the catalog that must be fetched for order items, with usage-based selling during the order creation process.

Usage

This enum is available when the Usage-Based Selling feature is enabled.

Enum Values

The `commerceorders.CatalogRatesPreferenceEnum` enum includes these values.

Value	Description
`Fetch`	Retrieves the rate card entries defined in the catalog for order items during the order creation process.
`Skip`	Skips the retrieval of rate card entries for order items during the order creation process. The default value is `Skip`.

ConfigurationInputEnum Enum

Specifies the configuration input for the request to place an order.

Usage

Use these enum values for the `configurationInputEnum` property in the PlaceOrderExecutor Class

Enum Values

The `commerceorders.ConfigurationInputEnum` enum has these values.

Value	Description
`RunAndAllowErrors`	Run the configuration and proceed with order ingestion upon encountering any configuration errors.
`RunAndBlockErrors`	Run the configuration and block order ingestion upon encountering any configuration errors.
`Skip`	Skip the configuration execution.

ConfigurationOptionsInput Class

Contains methods and properties to set the configuration options for the input to the product configurator.

Namespace

CommerceOrders

Usage

This class holds the required details of the product configuration input. Set the class properties to enable default configuration, execution of configuration rules, and validation of the product catalog. Use these class properties as an input to the PlaceOrderExecutor Class method.

Example

```
CommerceOrders.ConfigurationOptionsInput configurationInput = new
CommerceOrders.ConfigurationOptionsInput();
configurationInput.validateProductCatalog = true;
configurationInput.validateAmendRenewCancel = true;
configurationInput.executeConfigurationRules = true;
configurationInput.addDefaultConfiguration = true;

CommerceOrders.GraphRequest graph = new CommerceOrders.GraphRequest('testGraph',
recordNodes);
CommerceOrders.PlaceOrderResult result = CommerceOrders.PlaceOrderExecutor.execute(graph,
 pricingPreference, configurationPreference, configurationInput);
```

ConfigurationOptionsInput Properties

Set the `ConfigurationOptionsInput` class properties to add default configuration, execute configuration rules, and validate the product catalog.

ConfigurationOptionsInput Methods

Learn more about the methods available with the `ConfigurationOptionsInput` class.

ConfigurationOptionsInput Properties

Set the `ConfigurationOptionsInput` class properties to add default configuration, execute configuration rules, and validate the product catalog.

The `ConfigurationOptionsInput` class includes these properties.

addDefaultConfiguration
Sets the default product configuration, such as bundle and product attributes, for an order request.

executeConfigurationRules
Sets the requirement for an order to adhere to the configuration rules.

validateAmendRenewCancel
Sets the requirement to run validations related to amend, renew, or cancel processes.

validateProductCatalog
Sets the requirement to validate an order against the product catalog.

`addDefaultConfiguration`

Sets the default product configuration, such as bundle and product attributes, for an order request.

Signature

```
public Boolean addDefaultConfiguration {get; set;}
```

Property Value

Type: Boolean

Indicates whether to automatically add default configuration to the order (`true`) or not (`false`).

`executeConfigurationRules`

Sets the requirement for an order to adhere to the configuration rules.

Signature

```
public Boolean executeConfigurationRules {get; set;}
```

Property Value

Type: Boolean

Indicates whether the order must adhere to configuration rules during processing (`true`) or bypass them (`false`).

`validateAmendRenewCancel`

Sets the requirement to run validations related to amend, renew, or cancel processes.

Signature

```
public Boolean validateAmendRenewCancel {get; set;}
```

Property Value

Type: Boolean

Indicates whether to run validations related to amend, renew, or cancel processes (`true`) or not (`false`).

`validateProductCatalog`

Sets the requirement to validate an order against the product catalog.

Signature

```
public Boolean validateProductCatalog {get; set;}
```

Property Value

Type: Boolean

Indicates whether the order must be validated against the product catalog (`true`) or not (`false`).

ConfigurationOptionsInput Methods

Learn more about the methods available with the `ConfigurationOptionsInput` class.

The `ConfigurationOptionsInput` class includes these methods.

equals(obj)
: Determines the equality of external objects in a list. This method is dynamic and is based on the `equals()` method in Java.

hashCode()
: Determines the uniqueness of the external object records in a list.

toString()
: Converts a value to a string.

`equals(obj)`

Determines the equality of external objects in a list. This method is dynamic and is based on the `equals()` method in Java.

Signature

```
public Boolean equals(Object obj)
```

Parameters

`obj`

Type: Object

Reference object that's used to compare with the class object.

Return Value

Type: Boolean

Indicates if the class object is same as the reference object (`true`) or not (`false`).

`hashCode()`

Determines the uniqueness of the external object records in a list.

Signature

```
public Integer hashCode()
```

Return Value

Type: Integer

Integer hash code that represents the value of the object. Equal objects as per the `equals()` method must return the same hash code.

`toString()`

Converts a value to a string.

Signature

```
public String toString()
```

Return Value

Type: String

GraphRequest Class

Contains constructors and properties to set the graph ID and a list of records to be ingested. The list of records is specified in a key-value map format that contains the field values of an order.

Namespace

CommerceOrders

Example

Create the list of records to be ingested by using these steps.

- Create the list of records by constructing the `Map<String, Object>` map of field values of an order.

```
List<CommerceOrders.RecordWithReferenceRequest> recordNodes = new
List<CommerceOrders.RecordWithReferenceRequest>();

// Prepare for the Order
Map<String,Object> orderFieldValues = new Map<String,Object>();
orderFieldValues.put('Pricebook2Id', '01sDU0000000lEIYAY');
orderFieldValues.put('AccountId', '001DU000001nIPKYA2');
orderFieldValues.put('EffectiveDate', '2024-01-01');
```

- To create a record object from the field values, create an instance of the `RecordResource` class.

```
CommerceOrders.RecordResource orderRecord = new
CommerceOrders.RecordResource(Order.getSobjectType(), 'POST');
orderRecord.fieldValues = orderFieldValues;
```

- To associate the Record object with a reference identifier, create an instance of the `RecordWithReferenceRequest` class.

```
CommerceOrders.RecordWithReferenceRequest orderRecordNode = new
CommerceOrders.RecordWithReferenceRequest('refOrder', orderRecord);
recordNodes.add(orderRecordNode);

// Prepare for the App Usage Assignment
Map<String,Object> auaFieldValues = new Map<String,Object>();
auaFieldValues.put('AppUsageType', 'RevenueLifecycleManagement');
auaFieldValues.put('RecordId', '@{refOrder.id}');

CommerceOrders.RecordResource auaRecord = new
CommerceOrders.RecordResource(AppUsageAssignment.getSobjectType(), 'POST');
auaRecord.fieldValues = auaFieldValues;

CommerceOrders.RecordWithReferenceRequest auaRecordNode = new
CommerceOrders.RecordWithReferenceRequest('refAppTag', auaRecord);
recordNodes.add(auaRecordNode);

// Prepare for the Order Item
Map<String,Object> oiFieldValues = new Map<String,Object>();
oiFieldValues.put('OrderId', '@{refOrder.id}');
oiFieldValues.put('PricebookEntryId', '01uDU000000YPkIYAW');
oiFieldValues.put('Product2Id', '01tDU000000ESCSYA4');
oiFieldValues.put('Quantity', 2);
oiFieldValues.put('UnitPrice', 800);

CommerceOrders.RecordResource oiRecord = new
CommerceOrders.RecordResource(OrderItem.getSobjectType(), 'POST');
oiRecord.fieldValues = oiFieldValues;

CommerceOrders.RecordWithReferenceRequest oiRecordNode = new
CommerceOrders.RecordWithReferenceRequest('refOrderItem', oiRecord);
recordNodes.add(oiRecordNode);
```

- Invoke the Place Order Apex API.

Note: The `CatalogRatesPreferenceEnum` enum is available when the Usage-Based Selling feature is enabled.

```
// Invoke the Place Order Apex API
CommerceOrders.PricingPreferenceEnum pricingPreference =
CommerceOrders.PricingPreferenceEnum.System;
CommerceOrders.CatalogRatesPreferenceEnum catalogRatesPreference =
CommerceOrders.CatalogRatesPreferenceEnum.Fetch;
CommerceOrders.ConfigurationInputEnum configurationPreference =
```

```
CommerceOrders.ConfigurationInputEnum.RunAndAllowErrors;
CommerceOrders.ConfigurationOptionsInput configurationInput = new
CommerceOrders.ConfigurationOptionsInput();
configurationInput.validateProductCatalog = true;
configurationInput.validateAmendRenewCancel = true;
configurationInput.executeConfigurationRules = true;
configurationInput.addDefaultConfiguration = true;
```

- To contain all record objects, create an instance of the `GraphRequest` class.

```
CommerceOrders.GraphRequest graph = new CommerceOrders.GraphRequest('testGraph',
recordNodes);
CommerceOrders.PlaceOrderResult result = CommerceOrders.PlaceOrderExecutor.execute(graph,
 pricingPreference, catalogRatesPreference, configurationPreference, configurationInput);

// Process any error, if exists
if (!result.success) {
  List<ConnectApi.PlaceOrderErrorResponse> errors = result.responseError;
  for (ConnectApi.PlaceOrderErrorResponse error : errors) {
    System.debug(error.errorCode + ': ' + error.message);
  }
}
```

GraphRequest Constructors

Learn more about the available constructors with the `GraphRequest` class.

GraphRequest Properties

Learn more about the available properties with the `GraphRequest` class.

GraphRequest Constructors

Learn more about the available constructors with the `GraphRequest` class.

The `GraphRequest` class includes these constructors.

GraphRequest(graphId, records)

Creates an instance of the `GraphRequest` class to assign the graph ID and a list of records to be ingested.

`GraphRequest(graphId, records)`

Creates an instance of the `GraphRequest` class to assign the graph ID and a list of records to be ingested.

Signature

```
public GraphRequest(String graphId, List<commerceorders.RecordWithReferenceRequest>
records)
```

Parameters

graphId

Type: String

ID of the graph.

records

Type: List<commerceorders.RecordWithReferenceRequest>

List of records to be ingested.

GraphRequest Properties

Learn more about the available properties with the `GraphRequest` class.

The `GraphRequest` class includes these properties.

graphId

Set the `graphId` property to assign the ID value of the graph.

`graphId`

Set the `graphId` property to assign the ID value of the graph.

Signature

```
public String graphId {get; set;}
```

Property Value

Type: String

PlaceOrderExecutor Class

Contains methods to place an order with details of the graph request, pricing preferences, and configuration options.

Namespace

CommerceOrders

Example

```
CommerceOrders.PlaceOrderResult resp =
CommerceOrders.PlaceOrderExecutor.execute(graph,internalEnum,cEnum,cInput,catalogRatesPreference);
```

PlaceOrderExecutor Methods

Learn more about the methods available with the `PlaceOrderExecutor` class.

PlaceOrderExecutor Example Implementation

Place orders with integrated pricing, configuration, and validation, and manage them throughout their entire lifecycle. To place an order from Apex, refer to the example implementation of the `PlaceOrderExecutor` class.

PlaceOrderExecutor Methods

Learn more about the methods available with the `PlaceOrderExecutor` class.

The `PlaceOrderExecutor` class includes these methods.

execute(graphRequest, pricingPreferenceEnum, configurationInputEnum, configurationOptionsInput)
Use the method in the `PlaceOrderExecutor` class to execute the Place Order Apex API request by assigning the properties for graph request, pricing reference, and configuration options.

execute(graphRequest, pricingPreferenceEnum, catalogRatesPreference, configurationInputEnum, configurationOptionsInput)
Use the method in the `PlaceOrderExecutor` class to execute the Place Order Apex API request by assigning the properties for graph request, pricing reference, and configuration options. This method also includes the property to define fetching of rate card entries.

execute(graphRequest)
Use the method in the `PlaceOrderExecutor` class to execute the Place Order Apex API request by assigning the properties for graph request.

`execute(graphRequest, pricingPreferenceEnum, configurationInputEnum, configurationOptionsInput)`

Use the method in the `PlaceOrderExecutor` class to execute the Place Order Apex API request by assigning the properties for graph request, pricing reference, and configuration options.

Signature

```
public static commerceorders.PlaceOrderResult execute(commerceorders.GraphRequest
graphRequest, commerceorders.PricingPreferenceEnum pricingPreferenceEnum,
commerceorders.ConfigurationInputEnum configurationInputEnum,
commerceorders.ConfigurationOptionsInput configurationOptionsInput)
```

Parameters

`graphRequest`
Type: commerceorders.GraphRequest

The sObject graph values of the order payload to be ingested.

`pricingPreferenceEnum`
Type: commerceorders.PricingPreferenceEnum

Pricing preference during the order process.

`configurationInputEnum`
Type: commerceorders.ConfigurationInputEnum

Configuration input for the place order process.

`configurationOptionsInput`
Type: commerceorders.ConfigurationOptionsInput on page 763

Configuration options during the ingestion process.

Return Value

Type: commerceorders.PlaceOrderResult

`execute(graphRequest, pricingPreferenceEnum, catalogRatesPreference, configurationInputEnum, configurationOptionsInput)`

Use the method in the `PlaceOrderExecutor` class to execute the Place Order Apex API request by assigning the properties for graph request, pricing reference, and configuration options. This method also includes the property to define fetching of rate card entries.

Signature

```
public static commerceorders.PlaceOrderResult execute(commerceorders.GraphRequest
graphRequest, commerceorders.PricingPreferenceEnum pricingPreferenceEnum,
commerceorders.CatalogRatesPreferenceEnum catalogRatesPreferenceEnum,
commerceorders.ConfigurationInputEnum configurationInputEnum,
commerceorders.ConfigurationOptionsInput configurationOptionsInput)
```

Parameters

`graphRequest`
Type: commerceorders.GraphRequest

The sObject graph values of the order payload to be ingested.

`pricingPreferenceEnum`
Type: commerceorders.PricingPreferenceEnum

Pricing preference during the order process.

`catalogRatesPreference`
Type: commerceorders.CatalogRatesPreferenceEnum

The rate card entries defined in the catalog that must be fetched for order items, with usage-based pricing during the order creation process. The `CatalogRatesPreferenceEnum` enum is available when the Usage-Based Selling feature is enabled.

`configurationInputEnum`
Type: commerceorders.ConfigurationInputEnum

Configuration input for the place order process.

`configurationOptionsInput`
Type: commerceorders.ConfigurationOptionsInput on page 763

Configuration options during the ingestion process.

Return Value

Type: commerceorders.PlaceOrderResult

`execute(graphRequest)`

Use the method in the `PlaceOrderExecutor` class to execute the Place Order Apex API request by assigning the properties for graph request.

Signature

```
public static commerceorders.PlaceOrderResult execute(commerceorders.GraphRequest
graphRequest)
```

Parameters

graphRequest

Type: commerceorders.GraphRequest

The sObject graph values of the order payload to be ingested.

Return Value

Type: commerceorders.PlaceOrderResult

PlaceOrderExecutor Example Implementation

Place orders with integrated pricing, configuration, and validation, and manage them throughout their entire lifecycle. To place an order from Apex, refer to the example implementation of the `PlaceOrderExecutor` class.

Namespace

commerceorders

Usage

Customize this example to suit your requirements. Create the list of records to be ingested by using these steps. Replace the respective IDs with the values that are present in your org. For example, replace the value of `${Pricebook2Id}` field with the price book ID that's present in the org.

Example:

- Create the list of records by constructing the `Map<String, Object>` map of field values of an order.

```
List<CommerceOrders.RecordWithReferenceRequest> recordNodes = new
List<CommerceOrders.RecordWithReferenceRequest>();

// Prepare for the Order
Map<String,Object> orderFieldValues = new Map<String,Object>();
orderFieldValues.put('Pricebook2Id', '01sDU0000000lEIYAY');
orderFieldValues.put('AccountId', '001DU000001nIPKYA2');
orderFieldValues.put('EffectiveDate', '2024-01-01');
```

- To create a record object from the field values, create an instance of the `RecordResource` class.

```
CommerceOrders.RecordResource orderRecord = new
CommerceOrders.RecordResource(Order.getSobjectType(), 'POST');
orderRecord.fieldValues = orderFieldValues;
```

- To associate the Record object with a reference identifier, create an instance of the `RecordWithReferenceRequest` class.

```
CommerceOrders.RecordWithReferenceRequest orderRecordNode = new
CommerceOrders.RecordWithReferenceRequest('refOrder', orderRecord);
recordNodes.add(orderRecordNode);

// Prepare for the App Usage Assignment
```

```
Map<String,Object> auaFieldValues = new Map<String,Object>();
auaFieldValues.put('AppUsageType', 'RevenueLifecycleManagement');
auaFieldValues.put('RecordId', '@{refOrder.id}');

CommerceOrders.RecordResource auaRecord = new
CommerceOrders.RecordResource(AppUsageAssignment.getSobjectType(), 'POST');
auaRecord.fieldValues = auaFieldValues;

CommerceOrders.RecordWithReferenceRequest auaRecordNode = new
CommerceOrders.RecordWithReferenceRequest('refAppTag', auaRecord);
recordNodes.add(auaRecordNode);

// Prepare for the Order Item
Map<String,Object> oiFieldValues = new Map<String,Object>();
oiFieldValues.put('OrderId', '@{refOrder.id}');
oiFieldValues.put('PricebookEntryId', '01uDU000000YPkIYAW');
oiFieldValues.put('Product2Id', '01tDU000000ESCSYA4');
oiFieldValues.put('Quantity', 2);
oiFieldValues.put('UnitPrice', 800);

CommerceOrders.RecordResource oiRecord = new
CommerceOrders.RecordResource(OrderItem.getSobjectType(), 'POST');
oiRecord.fieldValues = oiFieldValues;

CommerceOrders.RecordWithReferenceRequest oiRecordNode = new
CommerceOrders.RecordWithReferenceRequest('refOrderItem', oiRecord);
recordNodes.add(oiRecordNode);
```

- Invoke the Place Order Apex API.

  Note: The `CatalogRatesPreferenceEnum` enum is available when the Usage-Based Selling feature is enabled.

```
// Invoke the Place Order Apex API
CommerceOrders.PricingPreferenceEnum pricingPreference =
CommerceOrders.PricingPreferenceEnum.System;
CommerceOrders.CatalogRatesPreferenceEnum catalogRatesPreference =
CommerceOrders.CatalogRatesPreferenceEnum.Fetch;
CommerceOrders.ConfigurationInputEnum configurationPreference =
CommerceOrders.ConfigurationInputEnum.RunAndAllowErrors;
CommerceOrders.ConfigurationOptionsInput configurationInput = new
CommerceOrders.ConfigurationOptionsInput();
configurationInput.validateProductCatalog = true;
configurationInput.validateAmendRenewCancel = true;
configurationInput.executeConfigurationRules = true;
configurationInput.addDefaultConfiguration = true;
```

- To contain all record objects, create an instance of the `GraphRequest` class.

```
CommerceOrders.GraphRequest graph = new CommerceOrders.GraphRequest('testGraph',
recordNodes);
CommerceOrders.PlaceOrderResult result =
```

```
CommerceOrders.PlaceOrderExecutor.execute(graph, pricingPreference,
catalogRatesPreference, configurationPreference, configurationInput);


// Process any error, if exists
if (!result.success) {
  List<ConnectApi.PlaceOrderErrorResponse> errors = result.responseError;
  for (ConnectApi.PlaceOrderErrorResponse error : errors) {
    System.debug(error.errorCode + ': ' + error.message);
  }
}
```

PlaceOrderResult Class

Contains properties to hold the response to the place order request.

Namespace

CommerceOrders

Example

```
CommerceOrders.PlaceOrderResult resp =
CommerceOrders.PlaceOrderExecutor.execute(graph,internalEnum,cEnum,cInput,catalogRatesPreference);
```

PlaceOrderResult Properties

Learn more about the available properties with the `PlaceOrderResult` class.

PlaceOrderResult Properties

Learn more about the available properties with the `PlaceOrderResult` class.

The `PlaceOrderResult` class includes these properties.

orderId

Get the ID of the order that's created after a successful request.

requestIdentifier

Get the request ID of the process to query the asynchronous status of the Place Order Apex API.

responseError

Get the list of errors encountered during the synchronous processing of the API request.

statusURL

Get the asynchronous status URL of the request, if available.

success

Get the request status of the synchronous part of the processing.

orderId

Get the ID of the order that's created after a successful request.

Signature

```
public String orderId {get; set;}
```

Property Value

Type: String

requestIdentifier

Get the request ID of the process to query the asynchronous status of the Place Order Apex API.

Signature

```
public String requestIdentifier {get; set;}
```

Property Value

Type: String

responseError

Get the list of errors encountered during the synchronous processing of the API request.

Signature

```
public List<commerceorders.PlaceOrderErrorResponse> responseError {get; set;}
```

Property Value

Type: List<ConnectApi.PlaceOrderErrorResponse>

statusURL

Get the asynchronous status URL of the request, if available.

Signature

```
public String statusURL {get; set;}
```

Property Value

Type: String

success

Get the request status of the synchronous part of the processing.

Signature

```
public Boolean success {get; set;}
```

Property Value

Type: Boolean

Indicates whether the synchronous part of the processing is successful (`true`) or not (`false`).

PricingPreferenceEnum Enum

Specifies the pricing preference during the create order process.

Usage

Used by the PlaceOrderExecutor class.

Enum Values

The `commerceorders.PricingPreferenceEnum` enum includes these values.

Value	Description
`Force`	Enforce pricing during the order ingestion process.
`Skip`	Skip pricing during the order ingestion process.
`System`	Determine whether a pricing calculation is required.

RecordResource Class

Contains constructors and properties to create a record object from field values of an order.

Namespace

CommerceOrders

Example

```
CommerceOrders.RecordResource orderRecord = new
CommerceOrders.RecordResource(Order.getSobjectType(), 'POST');
orderRecord.fieldValues = orderFieldValues;
```

RecordResource Constructors

Learn more about the available constructors with the `RecordResource` class.

RecordResource Properties

Learn more about the available properties with the `RecordResource` class.

RecordResource Constructors

Learn more about the available constructors with the `RecordResource` class.

The `RecordResource` class includes these constructors.

RecordResource(type, method, id)
Creates an instance of the `RecordResource` class to assign values to the fields of an order item by using the sObject type, API method, and order ID properties.

RecordResource(type, method)
Creates an instance of the `RecordResource` class to assign the values to the fields of an order item by using the sObject type and API method properties.

`RecordResource(type, method, id)`

Creates an instance of the `RecordResource` class to assign values to the fields of an order item by using the sObject type, API method, and order ID properties.

Signature

```
public RecordResource(Schema.SObjectType type, String method, Id id)
```

Parameters

type
Type: Schema.SObjectType

Object that's returned from the field describe result using the `getReferenceTo()` method or from the sObject describe result using the `getSObjectType()` method.

method
Type: String

Method for the API request, such as POST or PATCH.

id
Type: Id

ID of the order.

`RecordResource(type, method)`

Creates an instance of the `RecordResource` class to assign the values to the fields of an order item by using the sObject type and API method properties.

Signature

```
public RecordResource(Schema.SObjectType type, String method)
```

Parameters

type
Type: Schema.SObjectType

Object that's returned from the field describe result using the `getReferenceTo()` method or from the sObject describe result using the `getSObjectType()` method.

`method`
Type: String

Method for the API request, such as POST or PATCH.

RecordResource Properties

Learn more about the available properties with the `RecordResource` class.

The `RecordResource` class includes these properties.

fieldValues
Set the `fieldValues` property to assign values to the fields to update the order record.

id
Set the `id` property to assign the ID of the order record.

method
Set the `method` property to specify the API request method, such as POST or PATCH.

type
Set the `type` property to assign the object type that's returned from the field describe result using the `getReferenceTo()` method or from the sObject describe result using the `getSObjectType()` method.

`fieldValues`

Set the `fieldValues` property to assign values to the fields to update the order record.

Signature

```
public Map<String,ANY> fieldValues {get; set;}
```

Property Value

Type: List<Map<String,ANY>>

`id`

Set the `id` property to assign the ID of the order record.

Signature

```
public String id {get; set;}
```

Property Value

Type: String

`method`

Set the `method` property to specify the API request method, such as POST or PATCH.

Signature

```
public String method {get; set;}
```

Property Value

Type: String

type

Set the `type` property to assign the object type that's returned from the field describe result using the `getReferenceTo()` method or from the sObject describe result using the `getSObjectType()` method.

Signature

```
public Schema.SObjectType type {get; set;}
```

Property Value

Type: Schema.SObjectType

RecordWithReferenceRequest Class

Contains constructors and properties to associate a record object with a reference identifier.

Namespace

CommerceOrders

Example

```
CommerceOrders.RecordWithReferenceRequest orderRecordNode = new
CommerceOrders.RecordWithReferenceRequest('refOrder', orderRecord);
```

RecordWithReferenceRequest Constructors

Learn more about the available constructors with the `RecordWithReferenceRequest` class.

RecordWithReferenceRequest Properties

Learn more about the available properties with the `RecordWithReferenceRequest` class.

RecordWithReferenceRequest Constructors

Learn more about the available constructors with the `RecordWithReferenceRequest` class.

The `RecordWithReferenceRequest` class includes these constructors.

RecordWithReferenceRequest(referenceId, record)

Creates an instance of the `RecordWithReferenceRequest` class to associate a record object with a reference identifier by using the `referenceId` and `record` object properties.

`RecordWithReferenceRequest(referenceId, record)`

Creates an instance of the `RecordWithReferenceRequest` class to associate a record object with a reference identifier by using the `referenceId` and `record` object properties.

Signature

```
public RecordWithReferenceRequest(String referenceId, commerceorders.RecordResource
record)
```

Parameters

`referenceId`
Type: String

Reference ID that maps to the subrequest response and can be used to reference the response in subsequent subrequests. You can reference the `referenceId` in either the body or URL of a subrequest. Use this syntax to include a reference: `@{referenceId.FieldName}`. See referenceId property of a composite subrequest.

`record`
Type: commerceorders.RecordResource

Record object that's defined using the `RecordResource` class.

RecordWithReferenceRequest Properties

Learn more about the available properties with the `RecordWithReferenceRequest` class.

The `RecordWithReferenceRequest` class includes these properties.

record
Set the `record` property to specify the record object that's defined by using the `RecordResource` class.

referenceId
Set the `referenceId` property to specify the reference ID that maps to the subrequest response. This reference ID can be used to reference the response in subsequent subrequests.

`record`

Set the `record` property to specify the record object that's defined by using the `RecordResource` class.

Signature

```
public commerceorders.RecordResource record {get; set;}
```

Property Value

Type: commerceorders.RecordResource

`referenceId`

Set the `referenceId` property to specify the reference ID that maps to the subrequest response. This reference ID can be used to reference the response in subsequent subrequests.

Signature

```
public String referenceId {get; set;}
```

Property Value

Type: String

CommerceTax Namespace

Manage the communication between Salesforce and an external tax engine.

The `CommerceTax` namespace includes these classes.

AbstractTransactionResponse Class

Abstract class that contains methods for setting tax fields based on the external tax provider's response. Response classes that extend `AbstractTransactionResponse` inherit these methods.

AddressesResponse Class

Sets the tax address fields based on a response from the external tax engine. Contains setter methods for the Ship From, Ship To, and Sold To addresses.

AddressResponse Class

Contains a location code sent from the external tax engine.

AmountDetailsResponse Class

Sets tax amount fields based on a response from the external tax engine.

CalculateTaxRequest Class

Represents a request to an external tax engine to calculate tax. Extends the TaxTransactionRequest class and is the top-level request class.

CalculateTaxResponse Class

Sets the values of the tax transaction following a response from the external tax engine. Extends the AbstractTransactionResponse class and is the top-level response class.

CalculateTaxType Enum

Shows whether a tax calculation request is for estimated or actual tax.

ErrorResponse Class

Use to respond with an error after receiving errors from the PaymentGatewayAdapter methods of the CommercePayments namespace, such as request-forbidden responses, custom validation errors, or expired API tokens.

HeaderTaxAddressesRequest Class

Captures the address values that are applicable for the quote or order transaction.

ImpositionResponse Class

Stores details of tax impositions from the external tax engine.

JurisdictionResponse Class

Stores details from the external tax engine about the tax jurisdiction used in the tax calculation process. A tax jurisdiction represents a government entity that collects tax.

LineItemResponse Class

Response class that stores details of a list of one or more line items on which the tax engine has calculated tax.

LineTaxAddressesRequest Class
Stores details of the addresses applied per line item in a tax calculation request.

RequestType Enum
Shows the type of tax request made to the tax engine.

ResultCode Enum
Code that represents the results of a tax request made to the tax engine.

RuleDetailsResponse Class
Contains details about the tax rules used for tax calculation.

TaxAddressesRequest Class
Contains methods to get and set tax address values.

TaxAddressRequest Class
Contains address details used for tax calculation.

TaxApiException Class
Contains details about any exceptions during the tax calculation process. Extends the `ApexBaseException` class.

TaxCustomerDetailsRequest Class
Contains customer details used in tax calculation.

TaxDetailsResponse Class
Stores details of the tax values that an external tax engine calculates in response to a tax calculation request.

TaxEngineAdapter Interface
Retrieves information from the tax engine and evaluates the information to define tax details.

TaxEngineContext Class
Wrapper class that stores details about the type of a tax calculation request.

TaxLineItemRequest Class
Contains line item details of a tax request.

TaxSellerDetailsRequest Class
Contains tax code details used in the tax calculation request.

TaxTransactionRequest Class
Abstract class for storing customer details used in tax calculation and estimation requests.

TaxTransactionStatus Enum
Shows whether the tax transaction has been committed or uncommitted.

TaxTransactionType Enum
Shows whether the tax transaction is for a credit or debit transaction.

AbstractTransactionResponse Class

Abstract class that contains methods for setting tax fields based on the external tax provider's response. Response classes that extend `AbstractTransactionResponse` inherit these methods.

Namespace

CommerceTax

AbstractTransactionResponse Methods

Learn more about the methods for AbstractTransactionResponse class.

AbstractTransactionResponse Methods

Learn more about the methods for AbstractTransactionResponse class.

The `AbstractTransactionResponse` class includes these methods.

setAddresses(addresses)

Uses an instance of `AddressesResponse` to set the values of tax address fields.

setAmountDetails(amountDetails)

Uses an instance of `AmountDetailsResponse` to set tax amount fields such as exemption amount and tax amount.

setCurrencyIsoCode(currencyIsoCode)

Sets the currencyIsoCode field.

setDescription(dscptn)

Sets the Description field.

setDocumentCode(documentCode)

Sets the DocumentCode field. Document codes are often used to reference tax documents that the external tax engine uses in the tax calculation process. Document code acts as a unique link to chain-related transactions, such as amendment or refunds.

setEffectiveDate(effectiveDate)

Sets the EffectiveDate field. Effective Date fields are optional fields that store the date that a transaction takes effect. We provide these fields only for recordkeeping purposes – for example, if you must report an effective date to an external general ledger system. Salesforce doesn't use them to calculate any tax or payment values.

setLineItems(lineItems)

Uses an instance of the `LineItemResponse` class to set a list of line items. Each line item represents an item sent to an external tax engine for tax calculation.

setReferenceDocumentCode(referenceDocumentCode)

Sets the ReferenceDocumentCode field. Use this field to store the code of an additional document used in the tax calculation process. For example, use this field in case of a refund for a previously taxed purchase.

setReferenceEntityId(referenceEntityId)

Sets the ID of a reference entity. In Commerce Tax, a reference entity represents a record related to the items sent to the external tax engine for tax calculation. For example, if you sent order items for tax calculation, you could define the parent order as the reference entity.

setTaxTransactionId(taxTrxnId)

Sets the TaxTransactionId field using the ID of a tax transaction record. In Commerce Tax, a tax transaction record stores information about a specific tax calculation process.

setTransactionDate(transactionDate)

Sets the TransactionDate field.

`setAddresses(addresses)`

Uses an instance of `AddressesResponse` to set the values of tax address fields.

Signature

```
global void setAddresses(commercetax.AddressesResponse addresses)
```

Parameters

addresses
Type: AddressesResponse

Class that contains methods to set the Ship To, Ship From, and Sold To address information.

Return Value

Type: void

setAmountDetails(amountDetails)

Uses an instance of `AmountDetailsResponse` to set tax amount fields such as exemption amount and tax amount.

Signature

```
global void setAmountDetails(commercetax.AmountDetailsResponse amountDetails)
```

Parameters

amountDetails
Type: AmountDetailsResponse

Class that contains methods to set the tax exemption amount, tax amount, total amount, and total amount with tax.

Return Value

Type: void

setCurrencyIsoCode(currencyIsoCode)

Sets the currencyIsoCode field.

Signature

```
global void setCurrencyIsoCode(String currencyIsoCode)
```

Parameters

currencyIsoCode
Type: String

Three-letter ISO 4217 currency code associated with a tax object.

Return Value

Type: void

`setDescription(dscptn)`

Sets the Description field.

Signature

```
global void setDescription(String dscptn)
```

Parameters

`dscptn`
Type: String

Optional field for providing additional information about a record.

Return Value

Type: void

`setDocumentCode(documentCode)`

Sets the DocumentCode field. Document codes are often used to reference tax documents that the external tax engine uses in the tax calculation process. Document code acts as a unique link to chain-related transactions, such as amendment or refunds.

Signature

```
global void setDocumentCode(String documentCode)
```

Parameters

`documentCode`
Type: String

Code for a tax document used in the tax calculation process.

Return Value

Type: void

`setEffectiveDate(effectiveDate)`

Sets the EffectiveDate field. Effective Date fields are optional fields that store the date that a transaction takes effect. We provide these fields only for recordkeeping purposes – for example, if you must report an effective date to an external general ledger system. Salesforce doesn't use them to calculate any tax or payment values.

Signature

```
global void setEffectiveDate(Datetime effectiveDate)
```

Parameters

`effectiveDate`
Type: Datetime

Optional field that stores the date that a transaction takes effect.

Return Value

Type: void

`setLineItems(lineItems)`

Uses an instance of the `LineItemResponse` class to set a list of line items. Each line item represents an item sent to an external tax engine for tax calculation.

Signature

```
global void setLineItems(List<commercetax.LineItemResponse> lineItems)
```

Parameters

`lineItems`

Type: List<LineItemResponse>

A list of line items sent to an external tax engine for tax calculation.

Return Value

Type: void

`setReferenceDocumentCode(referenceDocumentCode)`

Sets the ReferenceDocumentCode field. Use this field to store the code of an additional document used in the tax calculation process. For example, use this field in case of a refund for a previously taxed purchase.

Signature

```
global void setReferenceDocumentCode(String referenceDocumentCode)
```

Parameters

`referenceDocumentCode`

Type: String

The code for a document used in the tax calculation process.

Return Value

Type: void

`setReferenceEntityId(referenceEntityId)`

Sets the ID of a reference entity. In Commerce Tax, a reference entity represents a record related to the items sent to the external tax engine for tax calculation. For example, if you sent order items for tax calculation, you could define the parent order as the reference entity.

Signature

```
global void setReferenceEntityId(String referenceEntityId)
```

Parameters

referenceEntityId
Type: String

ID of a record related to the items sent for tax calculation.

Return Value

Type: void

setTaxTransactionId(taxTrxnId)

Sets the TaxTransactionId field using the ID of a tax transaction record. In Commerce Tax, a tax transaction record stores information about a specific tax calculation process.

Signature

```
global void setTaxTransactionId(String taxTrxnId)
```

Parameters

taxTrxnId
Type: String

The ID of a tax transaction record in Commerce Tax.

Return Value

Type: void

setTransactionDate(transactionDate)

Sets the TransactionDate field.

Signature

```
global void setTransactionDate(Datetime transactionDate)
```

Parameters

transactionDate
Type: Datetime

Date that a tax transaction occurred.

Return Value

Type: void

AddressesResponse Class

Sets the tax address fields based on a response from the external tax engine. Contains setter methods for the Ship From, Ship To, and Sold To addresses.

Namespace

CommerceTax

Usage

Because `AddressesResponse` contains multiple addresses, we recommend using multiple instances of `AddressResponse` to set unique values for each address.

Example

This code sample represents a portion of the code used in a mock tax adapter. In this example, you create three `AddressResponse` classes, set their location codes, and pass them to the `Ship To`, `Ship From`, and `Sold To` setter methods in `AddressesResponse`. In an actual implementation, your `AddressResponse` classes already have a location code based on the response from the external tax engine.

```
commercetax.AddressesResponse addressesRes = new commercetax.AddressesResponse();

//AddressResponse containing ShipTo information
commercetax.AddressResponse shipToAddress = new commercetax.AddressResponse();
shipToAddress.setLocationCode('1234567');

//AddressResponse containing ShipFrom information
commercetax.AddressResponse shipFromAddress = new commercetax.AddressResponse();
shipFromAddress.setLocationCode('84720385');

//AddressResponse containing Sold To information
commercetax.AddressResponse soldToAddress = new commercetax.AddressResponse();
soldToAddress.setLocationCode('92381749');

//set values of addressesRes
addressesRes.setShipFrom(shipFromAddress);
addressesRes.setShipTo(shipToAddress);
addressesRes.setSoldTo(soldToAddress);
```

AddressesResponse Methods

Learn more about the methods for AddressesResponse class.

AddressesResponse Methods

Learn more about the methods for AddressesResponse class.

The `AddressesResponse` class includes these methods.

setShipFrom(shipFrom)

Sets the value of a ShipFrom address field.

setShipTo(shipTo)
Sets the value of a ShipTo address field.

setSoldTo(soldTo)
Sets the value of a SoldTo address field.

`setShipFrom(shipFrom)`

Sets the value of a ShipFrom address field.

Signature

```
global void setShipFrom(commercetax.AddressResponse shipFrom)
```

Parameters

`shipFrom`
Type: AddressResponse

A single address. Use this generic address parameter to store any type of address, such as Ship From, Ship To, and Sold To details. Users set the specific address in an `AddressResponse` instance and then pass that instance to the `AddressesResponse`'s `setShipTo()`, `setShipFrom()`, and `setSoldTo()` methods as needed.

Return Value

Type: void

`setShipTo(shipTo)`

Sets the value of a ShipTo address field.

Signature

```
global void setShipTo(commercetax.AddressResponse shipTo)
```

Parameters

`shipTo`
Type: AddressResponse

Stores a single address. This is a generic address parameter and can be used to store any type of address, such as Ship From, Ship To, and Sold To details. Users set the specific address in an `AddressResponse` instance and then pass that instance to the `AddressesResponse`'s `setShipTo()`, `setShipFrom()`, and `setSoldTo()` methods as needed.

Return Value

Type: void

`setSoldTo(soldTo)`

Sets the value of a SoldTo address field.

Signature

```
global void setSoldTo(commercetax.AddressResponse soldTo)
```

Parameters

soldTo

Type: AddressResponse

Stores a single address. This is a generic address parameter and can be used to store any type of address, such as Ship From, Ship To, Sold To details. Users set the specific address in an AddressResponse instance and then pass that instance to the `AddressesResponse`'s `setShipTo()`, `setShipFrom()`, and `setSoldTo()` methods as needed.

Return Value

Type: void

AddressResponse Class

Contains a location code sent from the external tax engine.

Namespace

CommerceTax

Usage

Use the `AddressResponse` class to set unique values for each address.

```
commercetax.AddressesResponse addressesRes = new commercetax.AddressesResponse();

//AddressResponse containing ShipTo information
commercetax.AddressResponse shipToAddress = new commercetax.AddressResponse();
shipToAddress.setLocationCode('1234567');

//AddressResponse containing ShipFrom information
commercetax.AddressResponse shipFromAddress = new commercetax.AddressResponse();
shipFromAddress.setLocationCode('84720385');

//AddressResponse containing Sold To information
commercetax.AddressResponse soldToAddress = new commercetax.AddressResponse();
soldToAddress.setLocationCode('92381749');

//set values of addressesRes
addressesRes.setShipFrom(shipFromAddress);
addressesRes.setShipTo(shipToAddress);
addressesRes.setSoldTo(soldToAddress);
```

AddressResponse Methods

Learn more about the available methods with the `AddressResponse` class.

AddressResponse Methods

Learn more about the available methods with the `AddressResponse` class.

The `AddressResponse` class includes these methods.

setLocationCode(locationCode)
Sets the value of a LocationCode field.

`setLocationCode(locationCode)`

Sets the value of a LocationCode field.

Signature

```
global void setLocationCode(String locationCode)
```

Parameters

`locationCode`
Type: String

A code that contains address information. This value can be passed to a method that sets the value of an address field.

Return Value

Type: void

AmountDetailsResponse Class

Sets tax amount fields based on a response from the external tax engine.

Namespace

CommerceTax

Example

In this example, an instance of `AmountDetailsResponse` class in a mock adapter calculates several tax amount fields. The `totalTax` and `totalAmount` parameters were defined in an instance of `LineItemResponse` class. The adapter then assigns the instance to `lineItemResponse`.

```
commercetax.AmountDetailsResponse amountResponse = new commercetax.AmountDetailsResponse();
amountResponse.setTotalAmountWithTax(totalTax+totalAmount);
amountResponse.setExemptAmount(0);
amountResponse.setTotalAmount(totalAmount);
amountResponse.setTaxAmount(totalTax);
lineItemResponse.setAmountDetails(amountResponse);
```

AmountDetailsResponse Methods
Learn more about the methods available from the `AmountDetailsResponse` class.

AmountDetailsResponse Methods

Learn more about the methods available from the `AmountDetailsResponse` class.

The following are methods for `AmountDetailsResponse`.

setExemptAmount(exemptAmount)
Sets the value of the ExemptAmount field.

setTaxAmount(taxAmount)
Sets the value of the TaxAmount field.

setTotalAmount(totalAmount)
Sets the value of the TotalAmount field.

setTotalAmountWithTax(totalAmtWithTax)
Sets the value of the TotalAmountWithTax field.

`setExemptAmount(exemptAmount)`

Sets the value of the ExemptAmount field.

Signature

```
global void setExemptAmount(Double exemptAmount)
```

Parameters

exemptAmount
Type: Double

The amount of a line item's total amount that's exempt from tax calculation.

Return Value

Type: void

`setTaxAmount(taxAmount)`

Sets the value of the TaxAmount field.

Signature

```
global void setTaxAmount(Double taxAmount)
```

Parameters

taxAmount
Type: Double

The calculated amount of tax for a line item.

Return Value

Type: void

`setTotalAmount(totalAmount)`

Sets the value of the TotalAmount field.

Signature

```
global void setTotalAmount(Double totalAmount)
```

Parameters

`totalAmount`
Type: Double

The total amount of a line item, excluding tax.

Return Value

Type: void

`setTotalAmountWithTax(totalAmtWithTax)`

Sets the value of the TotalAmountWithTax field.

Signature

```
global void setTotalAmountWithTax(Double totalAmtWithTax)
```

Parameters

`totalAmtWithTax`
Type: Double

The total amount of a line item combined with the calculated tax for that line item.

Return Value

Type: void

CalculateTaxRequest Class

Represents a request to an external tax engine to calculate tax. Extends the TaxTransactionRequest class and is the top-level request class.

Namespace

CommerceTax

Example

See TaxEngineAdapter Example Implementation for more details on how to access information from the `CalculateTaxRequest` class.

CalculateTaxRequest Constructors
Learn more about the constructors that are available with the `CalculateTaxRequest` class. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

CalculateTaxRequest Properties
Learn more about the available properties with the `CalculateTaxRequest` class.

CalculateTaxRequest Methods
Learn more about the available methods with the `CalculateTaxRequest` class.

CalculateTaxRequest Constructors

Learn more about the constructors that are available with the `CalculateTaxRequest` class. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

The `CalculateTaxRequest` class includes these constructors.

CalculateTaxRequest(taxType)
This constructor is intended for test usage only and throws an exception if used outside of the Apex test context.

`CalculateTaxRequest(taxType)`

This constructor is intended for test usage only and throws an exception if used outside of the Apex test context.

Signature

```
global CalculateTaxRequest(commercetax.CalculateTaxType taxType)
```

Parameters

taxType
Type: CalculateTaxType

Indicates whether the tax calculation is for estimated tax or actual tax.

CalculateTaxRequest Properties

Learn more about the available properties with the `CalculateTaxRequest` class.

The `CalculateTaxRequest` class includes these properties.

isCommit
Indicates whether the tax calculation has to be committed or reported to government authorities.

taxTransactionType
Shows whether the tax transaction is for a credit or debit transaction.

taxType
Shows whether the tax calculation is for estimated or actual tax wherein only actual tax can be submitted.

`isCommit`

Indicates whether the tax calculation has to be committed or reported to government authorities.

Signature

```
global Boolean isCommit {get; set;}
```

Property Value

Type: Boolean

`taxTransactionType`

Shows whether the tax transaction is for a credit or debit transaction.

Signature

```
global commercetax.TaxTransactionType taxTransactionType {get; set;}
```

Property Value

Type: TaxTransactionType

`taxType`

Shows whether the tax calculation is for estimated or actual tax wherein only actual tax can be submitted.

Signature

```
global commercetax.CalculateTaxType taxType {get; set;}
```

Property Value

Type: CalculateTaxType

CalculateTaxRequest Methods

Learn more about the available methods with the `CalculateTaxRequest` class.

The `CalculateTaxRequest` class includes these methods.

equals(obj)
Maintains the integrity of lists of type `CalculateTaxRequest` 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 `CalculateTaxRequest` by determining the uniqueness of the external object records in a list.

toString()

Converts a value to a string.

`equals(obj)`

Maintains the integrity of lists of type `CalculateTaxRequest` 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 `CalculateTaxRequest` by determining the uniqueness of the external object records in a list.

Signature

```
global Integer hashCode()
```

Return Value

Type: Integer

`toString()`

Converts a value to a string.

Signature

```
global String toString()
```

Return Value

Type: String

CalculateTaxResponse Class

Sets the values of the tax transaction following a response from the external tax engine. Extends the AbstractTransactionResponse class and is the top-level response class.

Namespace

CommerceTax

Example

```
if(requestType == commercetax.RequestType.CalculateTax){
             commercetax.calculatetaxtype type = request.taxtype;
             String docCode='';
             if(request.DocumentCode == 'simulateEmptyDocumentCode')
                 docCode = '';
             else if(request.DocumentCode != null)
                 docCode =request.DocumentCode;
            else if(request.ReferenceEntityId != null) docCode = request.ReferenceEntityId;

             else docCode =  String.valueOf(getRandomInteger(0,2147483647));
             commercetax.CalculateTaxResponse response = new
commercetax.CalculateTaxResponse();
             if(request.isCommit == true) {
                 response.setStatus(commercetax.TaxTransactionStatus.Committed);
             } else {
                 response.setStatus(commercetax.TaxTransactionStatus.Uncommitted);
             }
             response.setDocumentCode(docCode);
             response.setReferenceDocumentCode(request.referenceDocumentCode);
             response.setTaxType(type);
             response.setStatusDescription('statusDescription');
             if(request.sellerDetails.code == 'testSellerCode') {
                 response.setDescription('SellerCode fetched from TaxEngine entity');
             }
             else {
                 response.setDescription('description');
             }
            response.setEffectiveDate(system.now());
            if(request.transactionDate == null) {
               response.setTransactionDate(system.now());
            } else {
               response.setTransactionDate(request.transactionDate);
            }
            if(request.taxTransactionType == null) {
                response.setTaxTransactionType(commercetax.TaxTransactionType.Debit);
            } else {
                response.setTaxTransactionType(request.taxTransactionType);
            }
             if(request.currencyIsoCode == null || request.currencyIsoCode == '') {
                response.setCurrencyIsoCode('USD');
             } else {
                response.setCurrencyIsoCode(request.currencyIsoCode);
             }
```

```
            response.setReferenceEntityId(request.ReferenceEntityId);
}
```

CalculateTaxResponse Methods

Learn more about the available methods with the `CalculateTaxResponse` class.

CalculateTaxResponse Methods

Learn more about the available methods with the `CalculateTaxResponse` class.

The `CalculateTaxResponse` class includes these methods.

setAddresses(addresses)

Sets the value of the Addresses field using the addresses contained in an instance of the AddressesResponse class.

setAmountDetails(amountDetails)

Sets the value of the AmountDetails field using an instance of `AmountDetailsResponse`.

setCurrencyIsoCode(currencyIsoCode)

Sets the value of the CurrencyIsoCode field of the `CalculateTaxResponse` object.

setDescription(dscptn)

Sets the value of the Description field of the `CalculateTaxResponse` object.

setDocumentCode(documentCode)

Sets the value of the DocumentCode field of the `CalculateTaxResponse` object.

setEffectiveDate(effectiveDate)

Sets the value of the EffectiveDate field of the `CalculateTaxResponse` object.

setLineItems(lineItems)

Sets the value of the LineItems field of the `CalculateTaxResponse` object.

setReferenceDocumentCode(referenceDocumentCode)

Sets the value of the ReferenceDocumentCode field of the `CalculateTaxResponse` object.

setReferenceEntityId(referenceEntityId)

Sets the value of the ReferenceEntityId field of the `CalculateTaxResponse` object.

setStatus(status)

Sets the value of the Status field of the `CalculateTaxResponse` object.

setStatusDescription(statusDescription)

Sets the value of the StatusDescription field of the `CalculateTaxResponse` object.

setTaxTransactionId(taxTrxnId)

Sets the value of the TaxTransactionId field of the `CalculateTaxResponse` object.

setTaxTransactionType(taxTransactionType)

Sets the value of the TaxTransactionType field of the `CalculateTaxResponse` object.

setTaxType(taxType)

Sets the value of the TaxType field of the `CalculateTaxResponse` object.

setTransactionDate(transactionDate)

Sets the value of the TransactionDate field of the `CalculateTaxResponse` object.

`setAddresses(addresses)`

Sets the value of the Addresses field using the addresses contained in an instance of the AddressesResponse class.

Signature

```
global void setAddresses(commercetax.AddressesResponse addresses)
```

Parameters

`addresses`

Type: AddressesResponse

Contains Ship To, Ship From, and Sold To addresses.

Return Value

Type: void

`setAmountDetails(amountDetails)`

Sets the value of the AmountDetails field using an instance of `AmountDetailsResponse`.

Signature

```
global void setAmountDetails(commercetax.AmountDetailsResponse amountDetails)
```

Parameters

`amountDetails`

Type: AmountDetailsResponse

The tax amount details for a line item on which tax was calculated.

Return Value

Type: void

`setCurrencyIsoCode(currencyIsoCode)`

Sets the value of the CurrencyIsoCode field of the `CalculateTaxResponse` object.

Signature

```
global void setCurrencyIsoCode(String currencyIsoCode)
```

Parameters

`currencyIsoCode`

Type: String

Three-letter ISO 4217 currency code associated with a tax object.

Return Value

Type: void

`setDescription(dscptn)`

Sets the value of the Description field of the `CalculateTaxResponse` object.

Signature

```
global void setDescription(String dscptn)
```

Parameters

`dscptn`

Type: String

Optional description for providing more information about the calculate tax response.

Return Value

Type: void

`setDocumentCode(documentCode)`

Sets the value of the DocumentCode field of the `CalculateTaxResponse` object.

Signature

```
global void setDocumentCode(String documentCode)
```

Parameters

`documentCode`

Type: String

Code for a tax document that's created by the tax engine for the calculation process.

Return Value

Type: void

`setEffectiveDate(effectiveDate)`

Sets the value of the EffectiveDate field of the `CalculateTaxResponse` object.

Signature

```
global void setEffectiveDate(Datetime effectiveDate)
```

Parameters

effectiveDate
Type: Datetime

The date a tax calculation action takes effect. This parameter is optional and is provided only for recordkeeping purpose. Additionally, this parameter is used to determine the tax rates or rules and overrides the transaction date. For example, if the tax calculation request is placed on January 3 and the transaction date is January 1, you can set the effective date as January 1.

Return Value

Type: void

setLineItems(lineItems)

Sets the value of the LineItems field of the `CalculateTaxResponse` object.

Signature

```
global void setLineItems(List<commercetax.LineItemResponse> lineItems)
```

Parameters

lineItems
Type: List<LineItemResponse>

Response object that the tax adapter populates from the response of the external tax engine.

Return Value

Type: void

setReferenceDocumentCode(referenceDocumentCode)

Sets the value of the ReferenceDocumentCode field of the `CalculateTaxResponse` object.

Signature

```
global void setReferenceDocumentCode(String referenceDocumentCode)
```

Parameters

referenceDocumentCode
Type: String

Code for a reference document used in the tax calculation process.

Return Value

Type: void

`setReferenceEntityId(referenceEntityId)`

Sets the value of the ReferenceEntityId field of the `CalculateTaxResponse` object.

Signature

```
global void setReferenceEntityId(String referenceEntityId)
```

Parameters

`referenceEntityId`
Type: String

ID of an entity related to the line items submitted for tax calculation. For example, if order items were sent for tax calculation, you could use the ID of their parent order.

Return Value

Type: void

`setStatus(status)`

Sets the value of the Status field of the `CalculateTaxResponse` object.

Signature

```
global void setStatus(commercetax.TaxTransactionStatus status)
```

Parameters

`status`
Type: TaxTransactionStatus

Indicates whether a tax transaction has been committed.

Return Value

Type: void

`setStatusDescription(statusDescription)`

Sets the value of the StatusDescription field of the `CalculateTaxResponse` object.

Signature

```
global void setStatusDescription(String statusDescription)
```

Parameters

`statusDescription`
Type: String

Optional value for providing more information about a tax transaction's status.

Return Value

Type: void

setTaxTransactionId(taxTrxnId)

Sets the value of the TaxTransactionId field of the `CalculateTaxResponse` object.

Signature

```
public void setTaxTransactionId(String taxTrxnId)
```

Parameters

taxTrxnId
Type: String

The ID of the Salesforce tax transaction entity that stores information about the tax calculation transaction.

Return Value

Type: void

setTaxTransactionType(taxTransactionType)

Sets the value of the TaxTransactionType field of the `CalculateTaxResponse` object.

Signature

```
global void setTaxTransactionType(commercetax.TaxTransactionType taxTransactionType)
```

Parameters

taxTransactionType
Type: TaxTransactionType

Whether the tax transaction was for a credit or debit transacation.

Return Value

Type: void

setTaxType(taxType)

Sets the value of the TaxType field of the `CalculateTaxResponse` object.

Signature

```
global void setTaxType(commercetax.CalculateTaxType taxType)
```

Parameters

taxType
Type: CalculateTaxType

Indicates whether a tax calculation request is for estimated or actual tax.

Return Value

Type: void

setTransactionDate(transactionDate)

Sets the value of the TransactionDate field of the `CalculateTaxResponse` object.

Signature

```
global void setTransactionDate(Datetime transactionDate)
```

Parameters

transactionDate
Type: Datetime

The date that the tax transaction occurred.

Return Value

Type: void

CalculateTaxType Enum

Shows whether a tax calculation request is for estimated or actual tax.

Usage

Used by the CalculateTaxRequest and CalculateTaxResponse class methods.

Enum Values

The `commercetax.CalculateTaxType` enum includes these values.

Value	Description
`Actual`	Specifies that the tax calculation service should calculate the finalized (actual) tax for the requested line items.
`Estimated`	Specifies that the tax calculation service should estimate the tax for the requested line items.

ErrorResponse Class

Use to respond with an error after receiving errors from the PaymentGatewayAdapter methods of the CommercePayments namespace, such as request-forbidden responses, custom validation errors, or expired API tokens.

Namespace

CommerceTax

Example

This example snippet of a mock tax adapter shows a hypothetical scenario to demo an error response. The adapter receives request information from `TaxEngineContext` and stores it in an instance of `CalculateTaxRequest`. If the request's `documentCode` property is null or indicates an error, the adapter returns an error response with information about the error.

```
global virtual class MockAdapter implements commercetax.TaxEngineAdapter {

     global commercetax.TaxEngineResponse processRequest(commercetax.TaxEngineContext
taxEngineContext) {
         commercetax.RequestType requestType = taxEngineContext.getRequestType();
         commercetax.CalculateTaxRequest request =
(commercetax.CalculateTaxRequest)taxEngineContext.getRequest();

if(request.documentCode == null) {
             return new commercetax.ErrorResponse(commercetax.resultcode.TaxEngineError,
'404', 'documentCode  is mandatory');
         }
         if(request.documentCode == 'TaxEngineError') {
             return new commercetax.ErrorResponse(commercetax.resultcode.TaxEngineError,
'504', 'documentCode  - not supported');
         }
         if(request.documentCode == 'simulateValidationFailureInAdapter') {
             return new commercetax.ErrorResponse(commercetax.resultcode.TaxEngineError,
'400', 'validations for documentCode failed in adapter');
         }
         if(request.documentCode == 'simulateMalformedErrorInAdapter') {
                     return new
commercetax.ErrorResponse(commercetax.resultcode.TaxEngineError, null, 'malformed adapter
 error response');
         }
         if(request.documentCode == 'simulateTaxEngineProcessFailure') {
             return new commercetax.ErrorResponse(commercetax.resultcode.TaxEngineError,
'500', 'Tax Engine couldnt process your request');
         }
```

ErrorResponse Constructors

Learn more about the available constructors with the `ErrorResponse` class.

ErrorResponse Constructors

Learn more about the available constructors with the `ErrorResponse` class.

The `ErrorResponse` class includes these constructors.

ErrorResponse(resultCode, errorCode, errorMessage)
Constructor to initialize an `ErrorResponse` object from the result code, error code, and error message sent from the tax engine.

`ErrorResponse(resultCode, errorCode, errorMessage)`

Constructor to initialize an `ErrorResponse` object from the result code, error code, and error message sent from the tax engine.

Signature

```
global ErrorResponse(commercetax.ResultCode resultCode, String errorCode, String
errorMessage)
```

Parameters

resultCode
Type: ResultCode

Code for the type of result sent by the tax engine.

errorCode
Type: String

Code for the type of error sent by the tax engine.

Codes must match the HTTP status codes to be returned to the user. Here are a few examples:

- If the status code is for a bad request, set `errorCode` to `400`.
- If the status code is for a forbidden request, set `errorCode` to `403`.
- If `errorCode` isn't a valid HTTP status code, a 500 internal server error is returned.

errorMessage
Type: String

The error message sent by the tax engine.

HeaderTaxAddressesRequest Class

Captures the address values that are applicable for the quote or order transaction.

Namespace

CommerceTax

HeaderTaxAddressesRequest Constructors
Learn more about the constructors available with the `HeaderTaxAddressesRequest` class.

HeaderTaxAddressesRequest Properties
Learn more about the available properties with the `HeaderTaxAddressesRequest` class.

HeaderTaxAddressesRequest Methods
Learn more about the available methods with the `HeaderTaxAddressesRequest` class.

HeaderTaxAddressesRequest Constructors

Learn more about the constructors available with the `HeaderTaxAddressesRequest` class.

The `HeaderTaxAddressesRequest` class includes these constructors.

HeaderTaxAddressesRequest(shipFrom, shipTo, soldTo, billTo, taxEngineAddress)

Constructor for initializing the required addresses of the tax addresses request such as the ship from, ship to, sold to, and bill to addresses. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

`HeaderTaxAddressesRequest(shipFrom, shipTo, soldTo, billTo, taxEngineAddress)`

Constructor for initializing the required addresses of the tax addresses request such as the ship from, ship to, sold to, and bill to addresses. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

Signature

```
global HeaderTaxAddressesRequest(commercetax.TaxAddressRequest shipFrom,
commercetax.TaxAddressRequest shipTo, commercetax.TaxAddressRequest soldTo,
commercetax.TaxAddressRequest billTo, commercetax.TaxAddressRequest taxEngineAddress)
```

Parameters

shipFrom
Type: TaxAddressRequest

Address where a line item was shipped from.

shipTo
Type: TaxAddressRequest

Address where a line item was shipped to.

soldTo
Type: TaxAddressRequest

Address of the line item's buyer.

billTo
Type: TaxAddressRequest

Person or group who was billed for the line item.

taxEngineAddress
Type: TaxAddressRequest

Address that the tax engine uses to calculate tax.

HeaderTaxAddressesRequest Properties

Learn more about the available properties with the `HeaderTaxAddressesRequest` class.

The `HeaderTaxAddressesRequest` class includes these properties.

billTo

Specifies the billTo address for a line item on which tax was calculated.

shipFrom
Specifies the shipFrom address for a line item on which tax was calculated.

shipTo
Specifies the shipTo address for a line item on which tax was calculated.

soldTo
Specifies the soldTo address for a line item on which tax was calculated.

taxEngineAddress
Address used by the tax engine when calculating tax for a line item.

billTo

Specifies the billTo address for a line item on which tax was calculated.

Signature

```
global commercetax.TaxAddressRequest billTo {get; set;}
```

Property Value

Type: TaxAddressRequest

shipFrom

Specifies the shipFrom address for a line item on which tax was calculated.

Signature

```
global commercetax.TaxAddressRequest shipFrom {get; set;}
```

Property Value

Type: TaxAddressRequest

shipTo

Specifies the shipTo address for a line item on which tax was calculated.

Signature

```
global commercetax.TaxAddressRequest shipTo {get; set;}
```

Property Value

Type: TaxAddressRequest

soldTo

Specifies the soldTo address for a line item on which tax was calculated.

Signature

```
global commercetax.TaxAddressRequest soldTo {get; set;}
```

Property Value

Type: TaxAddressRequest

`taxEngineAddress`

Address used by the tax engine when calculating tax for a line item.

Signature

```
global commercetax.TaxAddressRequest taxEngineAddress {get; set;}
```

Property Value

Type: TaxAddressRequest

HeaderTaxAddressesRequest Methods

Learn more about the available methods with the `HeaderTaxAddressesRequest` class.

The `HeaderTaxAddressesRequest` class includes these methods.

equals(obj)
: Maintains the integrity of lists of type `HeaderTaxAddressesRequest` 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 `TaxAddressesRequest` by determining the uniqueness of the external objects in a list.

toString()
: Converts a value to a string.

`equals(obj)`

Maintains the integrity of lists of type `HeaderTaxAddressesRequest` 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 `TaxAddressesRequest` by determining the uniqueness of the external objects in a list.

Signature

```
global Integer hashCode()
```

Return Value

Type: Integer

`toString()`

Converts a value to a string.

Signature

```
global String toString()
```

Return Value

Type: String

ImpositionResponse Class

Stores details of tax impositions from the external tax engine.

Namespace

CommerceTax

Example

In this mock adapter example, the adapter sets the `TaxDetailsResponse.setImposition()` method parameter to null if the request's document code indicates that the tax calculation didn't require any exceptions. Otherwise, it creates an instance of `ImpositionResponse` and sets its SubType and Type values, and then assigns it to `TaxDetailsResponse`.

```
if(request.DocumentCode == 'SetsNullForResponseWithoutException'){
  taxDetailsResponse.setImposition(null);
                     }else{
                        commercetax.ImpositionResponse imposition = new
commercetax.ImpositionResponse();
                        imposition.setSubType('subtype');
                        imposition.setType('type');
                        taxDetailsResponse.setImposition(imposition);
                     }
```

ImpositionResponse Methods
Learn more about the available methods with the `ImpositionResponse` class.

ImpositionResponse Methods

Learn more about the available methods with the `ImpositionResponse` class.

The `ImpositionResponse` class includes these methods.

setId(id)
Sets the ID field of the `ImpositionResponse` class.

setName(name)
Sets the Name field of the `ImpositionResponse` class.

setSubType(subType)
Sets the SubType field of the `ImpositionResponse` class.

setType(type)
Sets the Type field of the `ImpositionResponse` class.

`setId(id)`

Sets the ID field of the `ImpositionResponse` class.

Signature

```
global void setId(String id)
```

Parameters

id
Type: String

User-defined ID value used for referencing the tax imposition.

Return Value

Type: void

`setName(name)`

Sets the Name field of the `ImpositionResponse` class.

Signature

```
global void setName(String name)
```

Parameters

name
Type: String

Optional user-defined name for the tax imposition response.

Return Value

Type: void

`setSubType(subType)`

Sets the SubType field of the `ImpositionResponse` class.

Signature

```
global void setSubType(String subType)
```

Parameters

`subType`
Type: String

Many tax calculation organizations use types and subtypes to categorize their tax imposition procedures. If the tax engine you use follows this process, set the subtype with this parameter.

Return Value

Type: void

`setType(type)`

Sets the Type field of the `ImpositionResponse` class.

Signature

```
public void setType(String type)
```

Parameters

`type`
Type: String

Many tax calculation organizations use types and subtypes to categorize their tax imposition procedures. If the tax engine you use follows this process, set the type with this parameter.

Return Value

Type: void

JurisdictionResponse Class

Stores details from the external tax engine about the tax jurisdiction used in the tax calculation process. A tax jurisdiction represents a government entity that collects tax.

Namespace

CommerceTax

Example

In this mock adapter example, the adapter sets the `TaxDetailsResponse.setJurisdiction()` method parameter to null if the request's document code indicates that the tax calculation didn't require any exceptions. Otherwise, it creates an instance of `JurisdictionResponse` and sets its address values. Because this code represents a mock adapter, the example defines the address parameters directly. In a standard implementation, the jurisdiction's setters receive values passed from the eternal tax engine.

```
if(request.DocumentCode == 'SetsNullForResponseWithoutException'){
                         taxDetailsResponse.setJurisdiction(null);
                     }else{
                        commercetax.JurisdictionResponse jurisdiction = new
commercetax.JurisdictionResponse();
                        jurisdiction.setCountry('country');
                        jurisdiction.setRegion('region');
                        jurisdiction.setName('name');
                        jurisdiction.setStateAssignedNumber('stateAssignedNo');
                        jurisdiction.setId('id');
                        jurisdiction.setLevel('level');
                        taxDetailsResponse.setJurisdiction(jurisdiction);

}
```

JurisdictionResponse Methods

Learn more about the available methods with the `JurisdictionResponse` class.

JurisdictionResponse Methods

Learn more about the available methods with the `JurisdictionResponse` class.

The `JurisdictionResponse` class includes these methods.

setCountry(country)

Sets the Country field of the `JurisdictionResponse` class.

setId(id)

Sets the ID field of the `JurisdictionResponse` class.

setLevel(level)

Sets the Level field of the `JurisdictionResponse` class.

setName(name)

Sets the Name field of the `JurisdictionResponse` class.

setRegion(region)

Sets the Region value of the `JurisdictionResponse` class.

setStateAssignedNumber(stateAssignedNo)

Sets the StateAssignedNumber field of the `JurisdictionResponse` class.

`setCountry(country)`

Sets the Country field of the `JurisdictionResponse` class.

Signature

```
global void setCountry(String country)
```

Parameters

`country`
Type: String

The country of the tax jurisdiction entity's address.

Return Value

Type: void

`setId(id)`

Sets the ID field of the `JurisdictionResponse` class.

Signature

```
global void setId(String id)
```

Parameters

`id`
Type: String

User-defined Id value used to reference the jurisdiction response.

Return Value

Type: void

`setLevel(level)`

Sets the Level field of the `JurisdictionResponse` class.

Signature

```
global void setLevel(String level)
```

Parameters

`level`
Type: String

Level value used in the jurisdiction entity's address.

Return Value

Type: void

setName(name)

Sets the Name field of the `JurisdictionResponse` class.

Signature

```
global void setName(String name)
```

Parameters

name
Type: String

Optional user-defined name field for referencing the jurisdiction response.

Return Value

Type: void

setRegion(region)

Sets the Region value of the `JurisdictionResponse` class.

Signature

```
global void setRegion(String region)
```

Parameters

region
Type: String

Region value used in the tax jurisdiction entity's address.

Return Value

Type: void

setStateAssignedNumber(stateAssignedNo)

Sets the StateAssignedNumber field of the `JurisdictionResponse` class.

Signature

```
global void setStateAssignedNumber(String stateAssignedNo)
```

Parameters

stateAssignedNo

Type: String

State assigned number value of the tax jurisdiction entity's address.

Return Value

Type: void

LineItemResponse Class

Response class that stores details of a list of one or more line items on which the tax engine has calculated tax.

Namespace

CommerceTax

Example

This example uses a `LineItemResponse` list to store information about each line item that was processed as part of the request. For simplicity, the sample code uses a static value of 1 for the tax rate. However, most integrations typically have a more complex process for determining a tax rate. Most integrations also build a `TaxDetailsResponse` list to store the actual tax value information that they assign to each line item in the `LineItemResponse` list.

```
Double totalTax = 0.0;
             Double totalAmount = 0.0;
             List<commercetax.LineItemResponse> lineItemResponses = new
List<commercetax.LineItemResponse>();
             for(Commercetax.TaxLineItemRequest lineItem : request.lineItems){
                 commercetax.AddressesResponse addressesRes = new
commercetax.AddressesResponse();
                 if(request.DocumentCode == 'SetsNullForResponseWithoutException'){
                     addressesRes.setShipFrom(null);
                     addressesRes.setShipTO(null);
                     addressesRes.setSoldTo(null);
                 }else{
                  commercetax.AddressResponse addRes = new commercetax.AddressResponse();

                     addRes.setLocationCode('locationCode');
                     addressesRes.setShipFrom(addRes);
                     addressesRes.setShipTO(addRes);
                     addressesRes.setSoldTo(addRes);
                 }
                 commercetax.LineItemResponse lineItemResponse = new
commercetax.LineItemResponse();
                 Double totalLineTax = 0;
                 List<commercetax.TaxDetailsResponse> taxDetailsResponses = new
List<commercetax.TaxDetailsResponse>();
                 for(integer i =0;i<1;i++){
                     Integer rate  = 1;
                     Double taxableAmount = lineItem.amount;
                     commercetax.TaxDetailsResponse taxDetailsResponse = new
```

```
commercetax.TaxDetailsResponse();
                    taxDetailsResponse.setRate(Double.valueOf(rate));
                    taxDetailsResponse.setTaxableAmount(taxableAmount);
                    Double tax = taxableAmount*rate;
                    totalLineTax+=tax;
                    taxDetailsResponse.setTax(taxableAmount*rate);
                    taxDetailsResponse.setExemptAmount(0);
                    taxDetailsResponse.setExemptReason('exemptReason');
                    taxDetailsResponse.setTaxRegionId('taxRegionId');

taxDetailsResponse.setTaxId(String.valueOf(getRandomInteger(0,2323233)));
                    taxDetailsResponse.setSerCode('serCode');
                    taxDetailsResponse.setTaxAuthorityTypeId('taxAuthorityTypeId');
                    if(request.DocumentCode == 'SetsNullForResponseWithoutException'){
                        taxDetailsResponse.setImposition(null);
                    }else{
                        commercetax.ImpositionResponse imposition = new
commercetax.ImpositionResponse();
                        imposition.setSubType('subtype');
                        imposition.setType('type');
                        taxDetailsResponse.setImposition(imposition);
                    }

                    if(request.DocumentCode == 'SetsNullForResponseWithoutException'){
                        taxDetailsResponse.setJurisdiction(null);
                    }else{
                        commercetax.JurisdictionResponse jurisdiction = new
commercetax.JurisdictionResponse();
                        jurisdiction.setCountry('country');
                        jurisdiction.setRegion('region');
                        jurisdiction.setName('name');
                        jurisdiction.setStateAssignedNumber('stateAssignedNo');
                        jurisdiction.setId('id');
                        jurisdiction.setLevel('level');
                        taxDetailsResponse.setJurisdiction(jurisdiction);
                    }

                    taxDetailsResponses.add(taxDetailsResponse);
                }
                lineItemResponse.setTaxes(taxDetailsResponses);
                totalTax +=totalLineTax;
                totalAmount+=lineItem.amount;
```

LineItemResponse Methods

Learn more about the available methods with the `LineItemResponse` class.

LineItemResponse Methods

Learn more about the available methods with the `LineItemResponse` class.

The `LineItemResponse` class includes these methods.

setAddresses(addresses)
Sets the Addresses field on the `LineItemResponse` using an instance of `AddressesResponse` class.

setAmountDetails(amountDetails)
Sets the Amount Details field on the `LineItemResponse` using an instance of `AmountDetails`.

setEffectiveDate(effectiveDate)
Sets the EffectiveDate field on the `LineItemResponse` class. Effective Date fields are optional fields that store the date that a transaction takes effect. We provide these fields only for recordkeeping purposes – for example, if you must report an effective date to an external general ledger system. Salesforce doesn't use them to calculate any tax or payment values.

setIsTaxable(isTaxable)
Sets the IsTaxable field on the `LineItemResponse` class.

setLineNumber(lineNumber)
Sets the LineNumber field on the `LineItemResponse` class.

setProductCode(productCode)
Sets the ProductCode field on the `LineItemResponse` class.

setQuantity(quantity)
Sets the Quantity field on the `LineItemResponse` class.

setTaxCode(taxCode)
Sets the TaxCode field on the `LineItemResponse`.

setTaxes(taxes)
Sets the Taxes field on a `LineItemResponse`.

`setAddresses(addresses)`

Sets the Addresses field on the `LineItemResponse` using an instance of `AddressesResponse` class.

Signature

```
global void setAddresses(commercetax.AddressesResponse addresses)
```

Parameters

addresses
Type: AddressesResponse

Class that contains methods to set the Ship To, Ship From, and Sold To address information.

Return Value

Type: void

`setAmountDetails(amountDetails)`

Sets the Amount Details field on the `LineItemResponse` using an instance of `AmountDetails`.

Signature

```
global void setAmountDetails(commercetax.AmountDetailsResponse amountDetails)
```

Parameters

amountDetails
Type: AmountDetailsResponse

Class that contains methods to set the tax amount, total amount with tax, total amount, and exempt amount.

Return Value

Type: void

setEffectiveDate(effectiveDate)

Sets the EffectiveDate field on the `LineItemResponse` class. Effective Date fields are optional fields that store the date that a transaction takes effect. We provide these fields only for recordkeeping purposes – for example, if you must report an effective date to an external general ledger system. Salesforce doesn't use them to calculate any tax or payment values.

Signature

```
global void setEffectiveDate(Datetime effectiveDate)
```

Parameters

effectiveDate
Type: Datetime

Optional field that stores the date that a transaction takes effect.

Return Value

Type: void

setIsTaxable(isTaxable)

Sets the IsTaxable field on the `LineItemResponse` class.

Signature

```
global void setIsTaxable(Boolean isTaxable)
```

Parameters

isTaxable
Type: Boolean

Whether line items were taxed as part of the tax calculation request.

Return Value

Type: void

`setLineNumber(lineNumber)`

Sets the LineNumber field on the `LineItemResponse` class.

Signature

```
global void setLineNumber(String lineNumber)
```

Parameters

`lineNumber`
Type: String

User-defined number used to identify a line item.

Return Value

Type: void

`setProductCode(productCode)`

Sets the ProductCode field on the `LineItemResponse` class.

Signature

```
global void setProductCode(String productCode)
```

Parameters

`productCode`
Type: String

Code for the product that a line item represents.

Return Value

Type: void

`setQuantity(quantity)`

Sets the Quantity field on the `LineItemResponse` class.

Signature

```
global void setQuantity(Double quantity)
```

Parameters

`quantity`
Type: Double

Quantity of a line item.

Return Value

Type: void

setTaxCode(taxCode)

Sets the TaxCode field on the `LineItemResponse`.

Signature

```
global void setTaxCode(String taxCode)
```

Parameters

taxCode

Type: String

Federal code that an individual or business uses to pay their taxes to a federal or state government. The tax engine uses this code during the tax calculation process.

Return Value

Type: void

setTaxes(taxes)

Sets the Taxes field on a `LineItemResponse`.

Signature

```
global void setTaxes(List<commercetax.TaxDetailsResponse> taxes)
```

Parameters

taxes

Type: List<TaxDetailsResponse>

Tax values applied to a line item in the `LineItemResponse` list. This information is stored in a list of `TaxDetailsResponses`, which contains values such as tax, taxable amount, and tax rate.

Return Value

Type: void

LineTaxAddressesRequest Class

Stores details of the addresses applied per line item in a tax calculation request.

Namespace

CommerceTax

LineTaxAddressesRequest Constructors

Learn more about the constructors available with the `LineTaxAddressesRequest` class.

LineTaxAddressesRequest Properties

Learn more about the available properties with the `LineTaxAddressesRequest` class.

LineTaxAddressesRequest Methods

Learn more about the available methods with the `LineTaxAddressesRequest` class.

LineTaxAddressesRequest Constructors

Learn more about the constructors available with the `LineTaxAddressesRequest` class.

The `LineTaxAddressesRequest` class includes these constructors.

LineTaxAddressesRequest(shipFrom, shipTo, soldTo, billTo, taxEngineAddress)

Constructor for initializing the required addresses for a line item of the tax addresses request such as the ship to, ship from, and bill to addresses. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

`LineTaxAddressesRequest(shipFrom, shipTo, soldTo, billTo, taxEngineAddress)`

Constructor for initializing the required addresses for a line item of the tax addresses request such as the ship to, ship from, and bill to addresses. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

Signature

```
global LineTaxAddressesRequest(commercetax.TaxAddressRequest shipFrom,
commercetax.TaxAddressRequest shipTo, commercetax.TaxAddressRequest soldTo,
commercetax.TaxAddressRequest billTo, commercetax.TaxAddressRequest taxEngineAddress)
```

Parameters

`shipFrom`
TaxAddressRequest

Address where a line item was shipped from.

`shipTo`
TaxAddressRequest

Address where a line item is shipped to.

`soldTo`
TaxAddressRequest

Address of the line item's buyer.

`billTo`
TaxAddressRequest

Person or group who was billed for the line item.

`taxEngineAddress`
TaxAddressRequest

Address that the tax engine uses to calculate tax.

LineTaxAddressesRequest Properties

Learn more about the available properties with the `LineTaxAddressesRequest` class.

The `LineTaxAddressesRequest` class includes these properties.

billTo
The Bill To address for a line item.

shipFrom
The Ship From address for a line item.

shipTo
The Ship To address for a line item.

soldTo
The Sold To address for a line item.

billTo

The Bill To address for a line item.

Signature

```
global commercetax.TaxAddressRequest billTo {get; set;}
```

Property Value

Type: TaxAddressRequest

shipFrom

The Ship From address for a line item.

Signature

```
global commercetax.TaxAddressRequest shipFrom {get; set;}
```

Property Value

Type: TaxAddressRequest

shipTo

The Ship To address for a line item.

Signature

```
global commercetax.TaxAddressRequest shipTo {get; set;}
```

Property Value

Type: TaxAddressRequest

soldTo

The Sold To address for a line item.

Signature

```
global commercetax.TaxAddressRequest soldTo {get; set;}
```

Property Value

Type: TaxAddressRequest

LineTaxAddressesRequest Methods

Learn more about the available methods with the `LineTaxAddressesRequest` class.

The `LineTaxAddressesRequest` class includes these methods.

equals(obj)
Maintains the integrity of lists of type `LineTaxAddressesRequest` 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 `LineTaxAddressesRequest` by determining the uniquness of the external object records in a list.

toString()
Converts a value to a string.

equals(obj)

Maintains the integrity of lists of type `LineTaxAddressesRequest` 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 `LineTaxAddressesRequest` by determining the uniquness of the external object records in a list.

Signature

```
global Integer hashCode()
```

Return Value

Type: Integer

`toString()`

Converts a value to a string.

Signature

```
global String toString()
```

Return Value

Type: String

RequestType Enum

Shows the type of tax request made to the tax engine.

Usage

Used by the TaxEngineContext class method.

Enum Values

The `commercetax.RequestType` enum includes these values.

Value	Description
`CalculateTax`	Represents a request to calculate tax on a list of taxable line items.

ResultCode Enum

Code that represents the results of a tax request made to the tax engine.

Usage

Used by the ErrorResponse class method.

Enum Values

The `commercetax.ResultCode` enum includes these values.

Value	Description
`TaxEngineError`	Represents an error that occurred during the tax request process.

RuleDetailsResponse Class

Contains details about the tax rules used for tax calculation.

Namespace

CommerceTax

RuleDetailsResponse Methods

Learn more about the available methods with the `RuleDetailsResponse` class.

RuleDetailsResponse Methods

Learn more about the available methods with the `RuleDetailsResponse` class.

The `RuleDetailsResponse` includes these methods.

RuleDetailsResponse()

Contains information about the tax rules used when calculating tax for line items.

setNonTaxableRuleId(nonTaxableRuleId)

Sets the NonTaxableRuleId field of the `RuleDetailsResponse`.

setNonTaxableType(nonTaxableType)

Sets the NonTaxableType field of the `RuleDetailsResponse`.

setRateRuleId(rateRuleId)

Sets the RateRuleId field of the `RuleDetailsResponse`.

setRateSourceId(rateSourceId)

Sets the RateSourceId field on the `RuleDetailsResponse`.

`RuleDetailsResponse()`

Contains information about the tax rules used when calculating tax for line items.

Signature

```
global void RuleDetailsResponse()
```

Return Value

Type: void

`setNonTaxableRuleId(nonTaxableRuleId)`

Sets the NonTaxableRuleId field of the `RuleDetailsResponse`.

Signature

```
global void setNonTaxableRuleId(String nonTaxableRuleId)
```

Parameters

nonTaxableRuleId
Type: String

ID of the tax rule applied to non-taxable line items.

Return Value

Type: void

setNonTaxableType(nonTaxableType)

Sets the NonTaxableType field of the `RuleDetailsResponse`.

Signature

```
global void setNonTaxableType(String nonTaxableType)
```

Parameters

nonTaxableType
Type: String

Reason (from several possible types) that a line item is non-taxable.

Return Value

Type: void

setRateRuleId(rateRuleId)

Sets the RateRuleId field of the `RuleDetailsResponse`.

Signature

```
global void setRateRuleId(String rateRuleId)
```

Parameters

rateRuleId
Type: String

ID of the tax rule used to determine a tax rate.

Return Value

Type: void

setRateSourceId(rateSourceId)

Sets the RateSourceId field on the `RuleDetailsResponse`.

Signature

```
global void setRateSourceId(String rateSourceId)
```

Parameters

rateSourceId
Type: String

ID of the source object used for calculating tax rate.

Return Value

Type: void

TaxAddressesRequest Class

Contains methods to get and set tax address values.

Namespace

CommerceTax

TaxAddressesRequest Constructors
Learn more about the available constructors with the `TaxAddressesRequest` class.

TaxAddressesRequest Properties
Learn more about the available properties with the `TaxAddressesRequest` class.

TaxAddressesRequest Methods
Learn more about the available methods with the `TaxAddressesRequest` class.

TaxAddressesRequest Constructors

Learn more about the available constructors with the `TaxAddressesRequest` class.

The `TaxAddressesRequest` class includes these constructors.

TaxAddressesRequest(shipFrom, shipTo, soldTo, billTo, taxEngineAddress)
Constructor for defining addresses for the tax addresses request. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

TaxAddressesRequest(shipFrom, shipTo, soldTo, billTo, taxEngineAddress)

Constructor for defining addresses for the tax addresses request. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

Signature

```
global TaxAddressesRequest(commercetax.TaxAddressRequest shipFrom,
commercetax.TaxAddressRequest shipTo, commercetax.TaxAddressRequest soldTo,
commercetax.TaxAddressRequest billTo, commercetax.TaxAddressRequest taxEngineAddress)
```

Parameters

shipFrom
TaxAddressRequest

The address where a line item was shipped from.

shipTo
TaxAddressRequest

The address where a line item is shipped to.

soldTo
TaxAddressRequest

The address of the line item's buyer.

billTo
TaxAddressRequest

The person or group who was billed for the line item.

taxEngineAddress
TaxAddressRequest

The address that the tax engine uses to calculate tax.

TaxAddressesRequest Properties

Learn more about the available properties with the `TaxAddressesRequest` class.

The `TaxAddressesRequest` class includes these properties.

billTo
The Bill To address for a line item.

shipFrom
The Ship From address for a line item.

shipTo
The Ship To address for a line item.

soldTo
The Sold To address for a line item.

taxEngineAddress
The Tax Engine Address for a line item.

billTo

The Bill To address for a line item.

Signature

```
global commercetax.TaxAddressRequest billTo {get; set;}
```

Property Value

TaxAddressRequest

shipFrom

The Ship From address for a line item.

Signature

```
global commercetax.TaxAddressRequest shipFrom {get; set;}
```

Property Value

TaxAddressRequest

shipTo

The Ship To address for a line item.

Signature

```
public commercetax.TaxAddressRequest shipTo {get; set;}
```

Property Value

TaxAddressRequest

soldTo

The Sold To address for a line item.

Signature

```
global commercetax.TaxAddressRequest soldTo {get; set;}
```

Property Value

TaxAddressRequest

taxEngineAddress

The Tax Engine Address for a line item.

Signature

```
global commercetax.TaxAddressRequest taxEngineAddress {get; set;}
```

Property Value

TaxAddressRequest

TaxAddressesRequest Methods

Learn more about the available methods with the `TaxAddressesRequest` class.

The `TaxAddressesRequest` class includes these methods.

equals(obj)

Maintains the integrity of lists of type `TaxAddressesRequest` 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 `TaxAddressesRequest` by determining the uniqueness of the external object records in a list.

toString()

Converts a value to a string.

`equals(obj)`

Maintains the integrity of lists of type `TaxAddressesRequest` 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 `TaxAddressesRequest` by determining the uniqueness of the external object records in a list.

Signature

```
global Integer hashCode()
```

Return Value

Type: Integer

`toString()`

Converts a value to a string.

Signature

```
global String toString()
```

Return Value

Type: String

TaxAddressRequest Class

Contains address details used for tax calculation.

Namespace

CommerceTax

TaxAddressRequest Constructors
Learn more about the available constructors with the `TaxAddressRequest` class.

TaxAddressRequest Properties
Learn more about the available properties with the `TaxAddressRequest` class.

TaxAddressRequest Methods
Learn more about the available methods with the `TaxAddressRequest` class.

TaxAddressRequest Constructors

Learn more about the available constructors with the `TaxAddressRequest` class.

The `TaxAddressRequest` class includes these constructors.

TaxAddressRequest(city, country, latitude, longitude, postalCode, state, street, locationCode)
Initializes the `TaxAddressRequest` object using address details. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

`TaxAddressRequest(city, country, latitude, longitude, postalCode, state, street, locationCode)`

Initializes the `TaxAddressRequest` object using address details. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

Signature

```
global TaxAddressRequest(String city, String country, Double latitude, Double longitude,
String postalCode, String state, String street, String locationCode)
```

Parameters

city
Type: String

City used in an address required for tax calculation.

country
Type: String

Country used in an address required for tax calculation.

latitude
Type: Double

Latitude used in an address required for tax calculation.

longitude
Type: Double

Longitude used in an address required for tax calculation.

postalCode
Type: String

Postal code used in an address required for tax calculation.

state
Type: String

State used in an address required for tax calculation.

street
Type: String

Street used in an address required for tax calculation.

locationCode
Type: String

Location code used in an address required for tax calculation.

TaxAddressRequest Properties

Learn more about the available properties with the `TaxAddressRequest` class.

The `TaxAddressRequest` class includes these properties.

city
City used in an address required for tax calculation.

country
Country used in an address required for tax calculation.

latitude
Latitude used in an address required for tax calculation.

locationCode
Location code used in an address required for tax calculation.

longitude
Longitude used in an address required for tax calculation.

postalCode
Postal code used in an address required for tax calculation.

state
State used in an address required for tax calculation.

street
Street used in an address required for tax calculation.

city

City used in an address required for tax calculation.

Signature

```
global String city {get; set;}
```

Property Value

Type: String

country

Country used in an address required for tax calculation.

Signature

```
global String country {get; set;}
```

Property Value

Type: String

latitude

Latitude used in an address required for tax calculation.

Signature

```
global Double latitude {get; set;}
```

Property Value

Type: Double

locationCode

Location code used in an address required for tax calculation.

Signature

```
global String locationCode {get; set;}
```

Property Value

Type: String

longitude

Longitude used in an address required for tax calculation.

Signature

```
global Double longitude {get; set;}
```

Property Value

Type: Double

postalCode

Postal code used in an address required for tax calculation.

Signature

```
global String postalCode {get; set;}
```

Property Value

Type: String

state

State used in an address required for tax calculation.

Signature

```
global String state {get; set;}
```

Property Value

Type: String

street

Street used in an address required for tax calculation.

Signature

```
global String street {get; set;}
```

Property Value

Type: String

TaxAddressRequest Methods

Learn more about the available methods with the `TaxAddressRequest` class.

The `TaxAddressRequest` class includes these methods.

equals(obj)
Maintains the integrity of lists of type TaxAddressRequest 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 `TaxAddressRequest` 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 TaxAddressRequest 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 `TaxAddressRequest` 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

TaxApiException Class

Contains details about any exceptions during the tax calculation process. Extends the `ApexBaseException` class.

Namespace

CommerceTax

TaxApiException Constructors

Learn more about the available constructors with the `TaxApiException` class.

TaxApiException Constructors

Learn more about the available constructors with the `TaxApiException` class.

The `TaxApiException` class includes these constructors.

TaxApiException(var1, var2)

Initializes the `TaxApiException` class using an `Exception` and a string to provide more details about the exception. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

TaxApiException(var1)

Initializes the `TaxApiException` class using an `Exception`. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

TaxApiException()

Initializes the `TaxApiException` class without any initialized parameters. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

`TaxApiException(var1, var2)`

Initializes the `TaxApiException` class using an `Exception` and a string to provide more details about the exception. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

Signature

```
global TaxApiException(String var1, Exception var2)
```

Parameters

var1

Type: String

Text that provides more information about the returned exception.

var2

Type: Exception

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.

`TaxApiException(var1)`

Initializes the `TaxApiException` class using an `Exception`. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

Signature

```
global TaxApiException(Exception var1)
```

Parameters

`var1`

Type: Exception

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.

`TaxApiException()`

Initializes the `TaxApiException` class without any initialized parameters. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

Signature

```
global TaxApiException()
```

TaxCustomerDetailsRequest Class

Contains customer details used in tax calculation.

Namespace

CommerceTax

TaxCustomerDetailsRequest Constructors

Learn more about the available constructors with the `TaxCustomerDetailsRequest` class.

TaxCustomerDetailsRequest Properties

Learn more about the available properties with the `TaxCustomerDetailsRequest` class.

TaxCustomerDetailsRequest Methods

Learn more about the available methods with the `TaxCustomerDetailsRequest` class.

TaxCustomerDetailsRequest Constructors

Learn more about the available constructors with the `TaxCustomerDetailsRequest` class.

The `TaxCustomerDetailsRequest` class includes these constructors.

TaxCustomerDetailsRequest(accountId, code, exemptionNo, exemptionReason)

Initializes the `TaxCustomerDetailsRequest` object. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

`TaxCustomerDetailsRequest(accountId, code, exemptionNo, exemptionReason)`

Initializes the `TaxCustomerDetailsRequest` object. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

Signature

```
global TaxCustomerDetailsRequest(String accountId, String code, String exemptionNo,
String exemptionReason)
```

Parameters

`accountId`

Type: String

The customer account ID for the line items sent for tax calculation.

`code`

Type: String

The tax code used during tax calculation.

`exemptionNo`

Type: String

The exemption number applied to any tax exempt line items.

`exemptionReason`

Type: String

The reason that certain line items are tax exempt.

TaxCustomerDetailsRequest Properties

Learn more about the available properties with the `TaxCustomerDetailsRequest` class.

The `TaxCustomerDetailsRequest` class includes these properties.

accountId

Customer account that contains the line items sent for tax calculation.

code

Tax code used during tax calculation.

exemptionNo

Number used to qualify a line item for tax exemption.

exemptionReason

Reason why a line item qualifies for tax exemption.

taxCertificateId

ID of a tax certificate used for tax calculation.

accountId

Customer account that contains the line items sent for tax calculation.

Signature

```
global String accountId {get; set;}
```

Property Value

Type: String

code

Tax code used during tax calculation.

Signature

```
global String code {get; set;}
```

Property Value

Type: String

exemptionNo

Number used to qualify a line item for tax exemption.

Signature

```
global String exemptionNo {get; set;}
```

Property Value

Type: String

exemptionReason

Reason why a line item qualifies for tax exemption.

Signature

```
global String exemptionReason {get; set;}
```

Property Value

Type: String

taxCertificateId

ID of a tax certificate used for tax calculation.

Signature

```
global String taxCertificateId {get; set;}
```

Property Value

Type: String

TaxCustomerDetailsRequest Methods

Learn more about the available methods with the `TaxCustomerDetailsRequest` class.

The `TaxCustomerDetailsRequest` class includes these methods.

equals(obj)
Maintains the integrity of lists of type `TaxCustomerDetailsRequest` 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 `TaxCustomerDetailsRequest` by determining the uniqueness of the external objects in a list.

toString()
Converts a value to a string.

`equals(obj)`

Maintains the integrity of lists of type `TaxCustomerDetailsRequest` 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 `TaxCustomerDetailsRequest` by determining the uniqueness of the external objects in a list.

Signature

```
global Integer hashCode()
```

Return Value

Type: Integer

`toString()`

Converts a value to a string.

Signature

```
global String toString()
```

Return Value

Type: String

TaxDetailsResponse Class

Stores details of the tax values that an external tax engine calculates in response to a tax calculation request.

Namespace

CommerceTax

Usage

If your tax calculation request contains multiple line items, we recommend building your adapter using a list of `TaxDetailsResponse` instances. Each instance represents the tax details calculated for a given line item.

Example

```
List<commercetax.TaxDetailsResponse> taxDetailsResponses = new
List<commercetax.TaxDetailsResponse>();
                 for(integer i =0;i<1;i++){
                     Integer rate  = 1;
                     Double taxableAmount = lineItem.amount;
                     commercetax.TaxDetailsResponse taxDetailsResponse = new
commercetax.TaxDetailsResponse();
                     taxDetailsResponse.setRate(Double.valueOf(rate));
                     taxDetailsResponse.setTaxableAmount(taxableAmount);
                     Double tax = taxableAmount*rate;
                     totalLineTax+=tax;
                     taxDetailsResponse.setTax(taxableAmount*rate);
                     taxDetailsResponse.setExemptAmount(0);
                     taxDetailsResponse.setExemptReason('exemptReason');
                     taxDetailsResponse.setTaxRegionId('taxRegionId');

taxDetailsResponse.setTaxId(String.valueOf(getRandomInteger(0,2323233)));
                     taxDetailsResponse.setSerCode('serCode');
                     taxDetailsResponse.setTaxAuthorityTypeId('taxAuthorityTypeId');
                     if(request.DocumentCode == 'SetsNullForResponseWithoutException'){
                         taxDetailsResponse.setImposition(null);
                     }else{
```

```
                        commercetax.ImpositionResponse imposition = new
commercetax.ImpositionResponse();
                        imposition.setSubType('subtype');
                        imposition.setType('type');
                        taxDetailsResponse.setImposition(imposition);
                    }

                    if(request.DocumentCode == 'SetsNullForResponseWithoutException'){
                        taxDetailsResponse.setJurisdiction(null);
                    }else{
                        commercetax.JurisdictionResponse jurisdiction = new
commercetax.JurisdictionResponse();
                        jurisdiction.setCountry('country');
                        jurisdiction.setRegion('region');
                        jurisdiction.setName('name');
                        jurisdiction.setStateAssignedNumber('stateAssignedNo');
                        jurisdiction.setId('id');
                        jurisdiction.setLevel('level');
                        taxDetailsResponse.setJurisdiction(jurisdiction);
                    }

                    taxDetailsResponses.add(taxDetailsResponse);
                }
                lineItemResponse.setTaxes(taxDetailsResponses);
                totalTax +=totalLineTax;
                totalAmount+=lineItem.amount;
```

TaxDetailsResponse Methods

Learn more about the available methods with the `TaxDetailsResponse` class.

TaxDetailsResponse Methods

Learn more about the available methods with the `TaxDetailsResponse` class.

The `TaxDetailsResponse` class includes these methods.

setExemptAmount(exemptAmount)

Sets the ExemptAmount field of the `TaxDetailsResponse` class.

setExemptReason(reason)

Sets the ExemptReason field of the `TaxDetailsResponse` class.

setImposition(imposition)

Sets the Imposition field of the `TaxDetailsResponse` class using an instance of the `ImpositionResponse` class.

setJurisdiction(jurisdiction)

Sets the Jurisdiction field of the `TaxDetailsResponse` using an instance of the `JurisdictionResponse` class.

setRate(rate)

Sets the Rate field of the `TaxDetailsResponse` class.

setSerCode(serCode)

Sets the Service Code field of the `TaxDetailsResponse` class.

setTax(tax)
Sets the Tax field of the `TaxDetailsResponse` class.

setTaxAuthorityTypeId(taxAuthorityTypeId)
Sets the TaxAuthorityTypeId field of the `TaxDetailsResponse` class.

setTaxId(taxId)
Sets the TaxId field of the `TaxDetailsResponse` class.

setTaxRegionId(taxRegionId)
Sets the TaxRegionId field on the `TaxDetailsResponse` class.

setTaxRuleDetails(taxRuleDetails)
Sets the TaxRuleDetails field of the `TaxDetailsResponse` class.

setTaxableAmount(taxableAmount)
Sets the TaxableAmount field of the `TaxDetailsResponse class`.

`setExemptAmount(exemptAmount)`

Sets the ExemptAmount field of the `TaxDetailsResponse` class.

Signature

```
global void setExemptAmount(Double exemptAmount)
```

Parameters

`exemptAmount`
Type: Double

Amount of tax on a line item that is exempt from tax calculation.

Return Value

Type: void

`setExemptReason(reason)`

Sets the ExemptReason field of the `TaxDetailsResponse` class.

Signature

```
global void setExemptReason(String reason)
```

Parameters

`reason`
Type: String

Optional user-defined information on why a tax exemption applies to a line item.

Return Value

Type: void

`setImposition(imposition)`

Sets the Imposition field of the `TaxDetailsResponse` class using an instance of the `ImpositionResponse` class.

Signature

```
global void setImposition(commercetax.ImpositionResponse imposition)
```

Parameters

`imposition`
Type: ImpositionResponse

Contains information about why tax was imposed on a line item.

Return Value

Type: void

`setJurisdiction(jurisdiction)`

Sets the Jurisdiction field of the `TaxDetailsResponse` using an instance of the `JurisdictionResponse` class.

Signature

```
global void setJurisdiction(commercetax.JurisdictionResponse jurisdiction)
```

Parameters

`jurisdiction`
Type: JurisdictionResponse

Contains address information about the tax jurisdiction used in the tax calculation process.

Return Value

Type: void

`setRate(rate)`

Sets the Rate field of the `TaxDetailsResponse` class.

Signature

```
global void setRate(Double rate)
```

Parameters

rate
Type: Double

Tax used during tax calculation. This value is often a decimal amount, such as 0.1 or 0.06, based on the applied tax percentage.

Return Value

Type: void

setSerCode(serCode)

Sets the Service Code field of the `TaxDetailsResponse` class.

Signature

```
global void setSerCode(String serCode)
```

Parameters

serCode
Type: String

Service code used in tax calculation.

Return Value

Type: void

setTax(tax)

Sets the Tax field of the `TaxDetailsResponse` class.

Signature

```
global void setTax(Double tax)
```

Parameters

tax
Type: Double

Amount of tax for a line item.

Return Value

Type: void

setTaxAuthorityTypeId(taxAuthorityTypeId)

Sets the TaxAuthorityTypeId field of the `TaxDetailsResponse` class.

Signature

```
global void setTaxAuthorityTypeId(String taxAuthorityTypeId)
```

Parameters

taxAuthorityTypeId
Type: String

ID of the organization that oversees tax collection.

Return Value

Type: void

setTaxId(taxId)

Sets the TaxId field of the `TaxDetailsResponse` class.

Signature

```
global void setTaxId(String taxId)
```

Parameters

taxId
Type: String

ID value used to determine the tax for an individual or business.

Return Value

Type: void

setTaxRegionId(taxRegionId)

Sets the TaxRegionId field on the `TaxDetailsResponse` class.

Signature

```
global void setTaxRegionId(String taxRegionId)
```

Parameters

taxRegionId
Type: String

ID of the tax region used in tax calculation. A tax region represents a geographical area where tax is applied.

Return Value

Type: void

`setTaxRuleDetails(taxRuleDetails)`

Sets the TaxRuleDetails field of the `TaxDetailsResponse` class.

Signature

```
global void setTaxRuleDetails(commercetax.RuleDetailsResponse taxRuleDetails)
```

Parameters

`taxRuleDetails`

Type: RuleDetailsResponse

Information about the Salesforce tax rules used during tax calculation.

Return Value

Type: void

`setTaxableAmount(taxableAmount)`

Sets the TaxableAmount field of the `TaxDetailsResponse class`.

Signature

```
global void setTaxableAmount(Double taxableAmount)
```

Parameters

`taxableAmount`

Type: Double

Amount that can be taxed on a line item.

Return Value

Type: void

TaxEngineAdapter Interface

Retrieves information from the tax engine and evaluates the information to define tax details.

Namespace

CommerceTax

TaxEngineAdapter Methods

Learn more about the available methods with the `TaxEngineAdapter` class.

TaxEngineAdapter Example Implementation

Refer to the example implementation of the `TaxEngineAdapter` interface to accept information from a tax engine and evaluate the information to define tax details.

TaxEngineAdapter Methods

Learn more about the available methods with the `TaxEngineAdapter` class.

The `TaxEngineAdapter` class includes these methods.

processRequest(requestType)
The `processRequest` method takes an instance of `TaxEngineContext` class and returns a response with the calculated tax details through the `TaxDetailsResponse` class or an error response through the `ErrorResponse` class.

`processRequest(requestType)`

The `processRequest` method takes an instance of `TaxEngineContext` class and returns a response with the calculated tax details through the `TaxDetailsResponse` class or an error response through the `ErrorResponse` class.

Signature

```
global commercetax.TaxEngineResponse processRequest(commercetax.TaxEngineContext var1)
```

Parameters

var1
Type: TaxEngineContext

Wrapper class that stores information about the type of a tax calculation request.

Return Value

Type: TaxEngineResponse

Generic interface representing a response from a tax engine.

TaxEngineAdapter Example Implementation

Refer to the example implementation of the `TaxEngineAdapter` interface to accept information from a tax engine and evaluate the information to define tax details.

Namespace

commercetax

Usage

The `TaxEngineAdapter` interface accepts information from the tax engine through the `TaxEngineContext` class. The interface evaluates the information to define tax in the response with details, such as tax amount and addresses. The response is used to update and create entities in the Salesforce org.

Use these steps to build a sample tax adapter implementation. Each tax adapter implementation varies based on your implementation requirements. Customize this example to suit your business requirements.

Example:

- The custom adapter class implements the `TaxEngineAdapter` interface. The `processRequest` method takes an instance of `TaxEngineContext` class and returns a response with the calculated tax details through the `TaxDetailsResponse` class or an error response through the `ErrorResponse` class.

```
global virtual class AvalaraAdapter implements commercetax.TaxEngineAdapter {
    global commercetax.TaxEngineResponse processRequest(commercetax.TaxEngineContext
 taxEngineContext) {
        commercetax.RequestType requestType = taxEngineContext.getRequestType();
        if(requestType == commercetax.RequestType.CalculateTax){
            return CalculateTaxService.getTax(taxEngineContext);
        }
        else
            return null;
    }
}
```

- This example shows the `CalculateTaxService` class.

```
global class CalculateTaxService {
    // ==============================================================================

    // CONSTANT
    // ==============================================================================

    private static final String AVALARA_ENDPOINT_URL_SANDBOX =
'https://sandbox-rest.avatax.com/api/v2';
    // Avalara Endpoint URL Production
    private static final String AVALARA_ENDPOINT_URL_PRODUCTION =
'https://rest.avatax.com/api/v2';
    private static final String TEST_REQUEST_BODY = '{  "id": -1,  "code": "00000131",
  "companyId": -1,  "date": "2017-02-03T00:00:00",  "taxDate": "2017-02-03T00:00:00",
  "status": "Temporary",  "type": "SalesOrder",  "reconciled": false,  "totalAmount":
 4000,  "totalExempt": 0,  "totalTax": 290,  "totalTaxable": 4000,
"totalTaxCalculated": 290,  "adjustmentReason": "NotAdjusted",  "locked": false,
"version": 1,  "modifiedDate": "2017-02-03T12:18:18.7347388Z",  "modifiedUserId":
53894,  "lines": [    {      "id": -1,      "transactionId": -1,      "lineNumber":
 "80241000000jNDCAA2",      "discountAmount": 0,      "exemptAmount": 0,
"exemptCertId": 0,      "isItemTaxable": true,      "lineAmount": 1000,
"reportingDate": "2017-02-03T00:00:00",      "tax": 72.5,      "taxableAmount":
1000,      "taxCalculated": 72.5,      "taxCode": "P0000000",      "taxDate":
"2017-02-03T00:00:00",      "taxIncluded": false,      "details": [        {
    "id": -1,          "transactionLineId": -1,          "transactionId": -1,
     "country": "US",          "region": "CA",          "exemptAmount": 0,
  "jurisCode": "06",          "jurisName": "CALIFORNIA",          "stateAssignedNo":
 "",          "jurisType": "STA",          "nonTaxableAmount": 0,          "rate":
 0.06,          "tax": 60,          "taxableAmount": 1000,          "taxType":
"Sales",          "taxName": "CA STATE TAX",          "taxAuthorityTypeId": 45,
       "taxCalculated": 60,          "rateType": "General"        },        {
     "id": -1,          "transactionLineId": -1,          "transactionId": -1,
      "country": "US",          "region": "CA",          "exemptAmount": 0,
   "jurisCode": "075",          "jurisName": "SAN FRANCISCO",
"stateAssignedNo": "",          "jurisType": "CTY",          "nonTaxableAmount": 0,
```

```
          "rate": 0.0025,          "tax": 2.5,          "taxableAmount": 1000,
      "taxType": "Sales",          "taxName": "CA COUNTY TAX",
"taxAuthorityTypeId": 45,          "taxCalculated": 2.5,          "rateType":
"General"        },        {          "id": -1,          "transactionLineId": -1,
         "transactionId": -1,          "country": "US",          "region": "CA",
       "exemptAmount": 0,          "jurisCode": "EMTV0",          "jurisName": "SAN
 FRANCISCO CO LOCAL TAX SL",          "stateAssignedNo": "38",          "jurisType":
 "STJ",          "nonTaxableAmount": 0,          "rate": 0.01,          "tax": 10,
          "taxableAmount": 1000,          "taxType": "Sales",          "taxName":
"CA SPECIAL TAX",          "taxAuthorityTypeId": 45,          "taxCalculated": 10,
          "rateType": "General"        }      ]    }  ]}';

    private static String getTestResponseString(){

     List<String> jsonResponse = new List<String> {
                                    '"id": 0',
                                    '"code": "testDocCode1231245984"',
                                    '"companyId": 468039',
                                    '"date": "2020-07-15"',
                                    '"paymentDate": "2020-07-15"',
                                    '"status": "Temporary"',
                                    '"type": "SalesOrder"',
                                    '"customerVendorCode": "testDocCode1234"',
                                    '"customerCode": "testDocCode1234"',
                                    '"reconciled": false',
                                    '"totalAmount": 232',
                                    '"totalExempt": 0',
                                    '"totalDiscount": 0',
                                    '"totalTax": 23.43',
                                    '"totalTaxable": 232',
                                    '"totalTaxCalculated": 23.43',
                                    '"adjustmentReason": "NotAdjusted"',
                                    '"locked": false',
                                    '"version": 1',
                                    '"exchangeRateEffectiveDate": "2020-07-15"',
                                    '"exchangeRate": 1',
                                  '"modifiedDate": "2020-08-13T11:19:20.4836636Z"',

                                    '"modifiedUserId": 53894',
                                    '"taxDate": "2020-07-15T00:00:00"',
                                    '"lines": [{"id": 0,"transactionId":
0,"lineNumber": "1","discountAmount": 0,"exemptAmount": 0,"exemptCertId":
0,"isItemTaxable": true,"itemCode": "","lineAmount": 232,"quantity":
1,"reportingDate": "2020-07-15","tax": 23.43,"taxableAmount": 232,"taxCalculated":
 23.43,"taxCode": "P0000000","taxCodeId": 8087,"taxDate":
"2020-07-15","taxOverrideType": "None","taxOverrideAmount": 0,"taxIncluded":
false,"details": [{"id": 0,"transactionLineId": 0,"transactionId": 0,"country":
"US","region": "WA","exemptAmount": 0,"jurisCode": "53","jurisName":
"WASHINGTON","stateAssignedNo": "","jurisType": "STA","jurisdictionType":
"State","nonTaxableAmount": 0,"rate": 0.065,"tax": 15.08,"taxableAmount":
232,"taxType": "Sales","taxSubTypeId": "S","taxName": "WA STATE
TAX","taxAuthorityTypeId": 45,"taxCalculated": 15.08,"rateType":
"General","rateTypeCode": "G","unitOfBasis": "PerCurrencyUnit","isNonPassThru":
false,"isFee": false},{"id": 0,"transactionLineId": 0,"transactionId": 0,"country":
```

```
 "US","region": "WA","exemptAmount": 0,"jurisCode": "033","jurisName":
"KING","stateAssignedNo": "1700","jurisType": "CTY","jurisdictionType":
"County","nonTaxableAmount": 0,"rate": 0,"tax": 0,"taxableAmount": 232,"taxType":
"Sales","taxSubTypeId": "S","taxName": "WA COUNTY TAX","taxAuthorityTypeId":
45,"taxCalculated": 0,"rateType": "General","rateTypeCode": "G","unitOfBasis":
"PerCurrencyUnit","isNonPassThru": false,"isFee": false}],"nonPassthroughDetails":
 [],"hsCode": "","costInsuranceFreight": 0,"vatCode": "","vatNumberTypeId": 0}]',
                                    '"addresses": [{"id": 0,"transactionId":
0,"boundaryLevel": "Address","line1": "255 S. King Street","line2": "","line3":
"","city": "Seattle","region": "WA","postalCode": "98104","country":
"US","taxRegionId": 2109700,"latitude": "47.59821","longitude": "-122.33108"}]',
                                    '"summary": [{"country": "US","region":
"WA","jurisType": "State","jurisCode": "53","jurisName":
"WASHINGTON","taxAuthorityType": 45,"stateAssignedNo": "","taxType":
"Sales","taxSubType": "S","taxName": "WA STATE TAX","rateType": "General","taxable":
 232,"rate": 0.065,"tax": 15.08,"taxCalculated": 15.08,"nonTaxable": 0,"exemption":
 0},{"country": "US","region": "WA","jurisType": "County","jurisCode":
"033","jurisName": "KING","taxAuthorityType": 45,"stateAssignedNo": "1700","taxType":
 "Sales","taxSubType": "S","taxName": "WA COUNTY TAX","rateType": "General","taxable":
 232,"rate": 0,"tax": 0,"taxCalculated": 0,"nonTaxable": 0,"exemption": 0}]'
                                };
            return '{' + String.join(jsonResponse, ',') + '}';
        }

    public static commercetax.TaxEngineResponse getTax(commercetax.TaxEngineContext
 taxEngineContext)
    {
        commercetax.CalculateTaxRequest request =
(commercetax.CalculateTaxRequest)taxEngineContext.getRequest();
        commercetax.calculatetaxtype requestType = request.taxtype;
        string referenceEntity = request.ReferenceEntityId;
        try{
            List<commercetax.TaxLineItemRequest> listOfLines = request.lineItems;
            if(!listOfLines.isEmpty()){
                HttpService sendHttpRequest = new HttpService();
                sendHttpRequest.addHeader('Content-type', 'application/json');
                String requestBody =
AvalaraJSONBuilder.getInstance().frameJsonForGetTaxOrderItem(request);
                sendHttpRequest.post('/transactions/create',requestBody);
                //system.debug('Request '+requestBody);
                String responseString = '';
                if(Test.isRunningTest()){
                    responseString = getTestResponseString();
                } else{
                    responseString = sendHttpRequest.getResponse().getBody();
                }
                //system.debug(sendHttpRequest.getResponse());
                //system.debug('response'+responseString);
                //responseString = TEST_REQUEST_BODY;
                system.debug('Heap size used ' +Limits.getHeapSize());

                if(!responseString.contains('error'))
                {
                    commercetax.CalculateTaxResponse response = new
```

```
commercetax.CalculateTaxResponse();
                    JsonSuccessParser jsonSuccessParserClass =
JsonSuccessParser.parse(responseString);
                    response.setTaxTransactionType(request.taxTransactionType);
                    response.setDocumentCode(jsonSuccessParserClass.code);

response.setReferenceDocumentCode(jsonSuccessParserClass.referenceCode);
                    if(jsonSuccessParserClass.status == 'Temporary')  {

response.setStatus(commercetax.TaxTransactionStatus.Uncommitted);
                    }
                    if(jsonSuccessParserClass.status == 'Committed') {

response.setStatus(commercetax.TaxTransactionStatus.Committed);
                    }
                    response.setTaxType(requestType);
                    commercetax.AmountDetailsResponse headerAmountResponse = new
commercetax.AmountDetailsResponse();

headerAmountResponse.setTotalAmountWithTax(jsonSuccessParserClass.totalAmount +
jsonSuccessParserClass.totaltax);

headerAmountResponse.setExemptAmount(jsonSuccessParserClass.totalExempt);

headerAmountResponse.setTotalAmount(jsonSuccessParserClass.totalAmount);

headerAmountResponse.setTaxAmount(jsonSuccessParserClass.totalTax);
                    response.setAmountDetails(headerAmountResponse);

response.setStatusDescription(jsonSuccessParserClass.adjustmentReason);

response.setEffectiveDate(date.valueof(jsonSuccessParserClass.taxDate));

response.setTransactionDate(date.valueof(jsonSuccessParserClass.transactionDate));
                    response.setReferenceEntityId(referenceEntity);
                    response.setTaxTransactionId(jsonSuccessParserClass.id);
                    response.setCurrencyIsoCode(request.currencyIsoCode);
                    List<commercetax.LineItemResponse> lineItemResponses = new
List<commercetax.LineItemResponse>();
                    for(JsonSuccessParser.Lines linesToProcess:
jsonSuccessParserClass.lines)
                    {
                        commercetax.LineItemResponse lineItemResponse = new
commercetax.LineItemResponse();
                        Double rateCalculated = 0.0;
                        List<commercetax.TaxDetailsResponse> taxDetailsResponses =
 new List<commercetax.TaxDetailsResponse>();
                        for(JsonSuccessParser.details linesDetails :
linesToProcess.details)
                        {
                            commercetax.TaxDetailsResponse taxDetailsResponse = new
 commercetax.TaxDetailsResponse();
                             if(linesDetails.exemptAmount != 0){
```

```
taxDetailsResponse.setExemptAmount(linesDetails.exemptAmount);
                                taxDetailsResponse.setExemptReason('Some reason we
 dont know');
                            }
                                commercetax.ImpositionResponse imposition = new
commercetax.ImpositionResponse();
                                    imposition.setSubType(linesDetails.taxName);
                                    imposition.setType(linesDetails.ratetype);
                                    imposition.setSubType(linesDetails.taxName);
                                    taxDetailsResponse.setImposition(imposition);
                                commercetax.JurisdictionResponse jurisdiction = new
 commercetax.JurisdictionResponse();
                                    jurisdiction.setCountry(linesDetails.country);
                                    jurisdiction.setRegion(linesDetails.region);
                                    jurisdiction.setName(linesDetails.jurisName);

jurisdiction.setStateAssignedNumber(linesDetails.stateAssignedNo);
                                    jurisdiction.setId(linesDetails.jurisCode);
                                    jurisdiction.setLevel(linesDetails.jurisType);
                                 taxDetailsResponse.setJurisdiction(jurisdiction);

                                    rateCalculated += linesDetails.rate;
                                taxDetailsResponse.setRate(rateCalculated);
                            taxDetailsResponse.setTax(linesDetails.taxCalculated);


taxDetailsResponse.setTaxableAmount(linesDetails.taxableAmount);

taxDetailsResponse.setTaxAuthorityTypeId(String.valueOf(linesDetails.taxAuthorityTypeId));

                                taxDetailsResponse.setTaxId(linesDetails.id);

taxDetailsResponse.setTaxRegionId(linesDetails.region);
                                taxDetailsResponses.add(taxDetailsResponse);

                        }
                            lineItemResponse.setTaxes(taxDetailsResponses);

lineItemResponse.setEffectiveDate(date.valueof(linesToProcess.taxDate));
                            lineItemResponse.setIsTaxable(true);
                                commercetax.AmountDetailsResponse amountResponse =
 new commercetax.AmountDetailsResponse();

amountResponse.setTaxAmount(linesToProcess.taxCalculated);

amountResponse.setTotalAmount(linesToProcess.lineAmount);

amountResponse.setTotalAmountWithTax(linesToProcess.lineAmount+linesToProcess.taxCalculated);


amountResponse.setExemptAmount(linesToProcess.exemptAmount);
                                lineItemResponse.setAmountDetails(amountResponse);

lineItemResponse.setIsTaxable(linesToProcess.isItemTaxable);
```

```
                        lineItemResponse.setProductCode(linesToProcess.itemCode);

                           lineItemResponse.setTaxCode(linesToProcess.taxCode);
                       lineItemResponse.setLineNumber(linesToProcess.lineNumber);

                           lineItemResponse.setQuantity(linesToProcess.quantity);
                           lineItemResponses.add(lineItemResponse);
                    }
                    response.setLineItems(lineItemResponses);
                    return response;
                }
                else
                {
                    JsonErrorParser jsonErrorParserClass =
JsonErrorParser.parse(responseString);
                    String message = null;
                    if(String.isNotBlank(jsonErrorParserClass.error.message))
                    {
                       message=jsonErrorParserClass.error.message;
                    }else{
                           String errorMessage = '';
                            for(JsonErrorParser.cls_details messageString :
jsonErrorParserClass.error.details)
                            {
                                if(String.isNotBlank(messageString.message) )
                                {
                                    errorMessage = messageString.message;
                                }
                            }
                            message = errorMessage;
                        }
                     return new
commercetax.ErrorResponse(commercetax.resultcode.TaxEngineError, '501', message);

                }
            }else return null;
        }
        catch (Exception e)
        {
            throw e;
        }
    }
}
```

- In the `HttpService` class, replace the `test` value in the endpoint variable with the name of the `TaxTypedNamedCredential` record. This class contains the credentials that are required to access your Avalara account through Salesforce.

```
public with sharing class HttpService
{
    // Attribute to implement singleton pattern for Order Product Service class
    private static HttpService httpServiceInstance;
```

```
    // VARIABLES

    private HttpResponse httpResponse;
    private Map<String,String> mapOfHeaderParameter = new Map<String,String>();
    private enum Method {GET, POST}

    /**
    * @name getInstance
    * @description get an Instance of Service class
    * @params NA
    * @return Http Service Class Instance
    */
    public static HttpService getInstance()
    {
        if (NULL == httpServiceInstance)
        {
            httpServiceInstance =  new HttpService();
        }
        return httpServiceInstance;
    }

    /**
    * @name get
    * @description Get Method to get a HTTP request
    */
    public void get(String endPoint)
    {
        send(newRequest(Method.GET, endPoint));
    }

    /**
    * @name post
    * @description Post Method to Post a HTTP request
    */
    public void post(String path, String requestBody)
    {
       String endPoint = 'callout:commerce.tax.TaxTypedNamedCredential:test'+path;
        send(newRequest(Method.POST, endPoint, requestBody));
    }

    /**
    * @name addHeader
    * @description addHeader Methods to add all the defualt Header's required fo
rthe request
    */
    public void addHeader(String name, String value)
    {
        mapOfHeaderParameter.put(name, value);
    }

    /**
    * @name setHeader
    * @description setHeader Methods to set setHeader for the request
    */
```

```
    private void setHeader(HttpRequest request)
    {
        for(String headerValue : mapOfHeaderParameter.keySet())
        {
            request.setHeader(headerValue, mapOfHeaderParameter.get(headerValue));
        }
    }
    /**
    * @name newRequest
    * @description newRequest Methods to make a new request
    */
    private HttpRequest newRequest(Method method, String endPoint)
    {
        return newRequest(method, endPoint, NULL);
    }

    /**
    * @name newRequest
    * @description newRequest Methods to make a new request
    */
   private HttpRequest newRequest(Method method, String endPoint, String requestBody)

    {
        HttpRequest request = new HttpRequest();
        request.setMethod(Method.name());
        setHeader(request);
        request.setEndpoint(endPoint);
        if (String.isNotBlank(requestBody))
        {
            request.setBody(requestBody);
        }
        request.setTimeout(120000);
        return request;
    }

    /**
    * @name send
    * @description send Methods to send a request
    */
    private void send(HttpRequest request)
    {
        try
        {
            Http http = new Http();
            httpResponse = http.send(request);
        }
        catch(System.CalloutException e)
        {
            system.debug('callout exception happened' + e.getMessage());
        }
        catch(Exception e)
        {
            system.debug('callout did not happen' + e.getMessage());
        }
```

```
    }

    /**
    * @name getResponse
    * @description getResponse Method to get the Response
    */
    public HttpResponse getResponse()
    {
        return httpResponse;
    }

    /**
    * @name getResponseToString
    * @description getResponse Method to get the Responses
    */
    public String getResponseToString()
    {
        return getResponse().toString();
    }
}
```

- Parse the `JsonSuccessParser` response object by using the `AvalaraJSONBuilder` class to build the response for your adapter.

  This example shows the `JsonSuccessParser` class.

```
global with sharing class JsonSuccessParser
{
  public static void consumeObject(JSONParser parser)
  {
    Integer depth = 0;
    do {
      JSONToken curr = parser.getCurrentToken();
      if (curr == JSONToken.START_OBJECT ||
        curr == JSONToken.START_ARRAY) {
        depth++;
      } else if (curr == JSONToken.END_OBJECT ||
        curr == JSONToken.END_ARRAY) {
        depth--;
      }
    } while (depth > 0 && parser.nextToken() != null);
  }

    public class Addresses {
        public String id {get;set;}
        public String transactionId {get;set;}
        public String boundaryLevel {get;set;}
        public String line1 {get;set;}
        public String city {get;set;}
        public String region {get;set;}
        public String postalCode {get;set;}
        public String country {get;set;}
        public Integer taxRegionId {get;set;}

        public Addresses(JSONParser parser) {
```

```
        while (parser.nextToken() != JSONToken.END_OBJECT) {
            if (parser.getCurrentToken() == JSONToken.FIELD_NAME) {
                String text = parser.getText();
                if (parser.nextToken() != JSONToken.VALUE_NULL) {
                    if (text == 'id') {
                        id = parser.getText();
                    } else if (text == 'transactionId') {
                        transactionId = parser.getText();
                    } else if (text == 'boundaryLevel') {
                        boundaryLevel = parser.getText();
                    } else if (text == 'line1') {
                        line1 = parser.getText();
                    } else if (text == 'city') {
                        city = parser.getText();
                    } else if (text == 'region') {
                        region = parser.getText();
                    } else if (text == 'postalCode') {
                        postalCode = parser.getText();
                    } else if (text == 'country') {
                        country = parser.getText();
                    } else if (text == 'taxRegionId') {
                        taxRegionId = parser.getIntegerValue();
                    } else {
                        consumeObject(parser);
                    }
                }
            }
        }
    }
}

public class Details {
    public String id {get;set;}
    public String transactionLineId {get;set;}
    public String transactionId {get;set;}
    public String country {get;set;}
    public String region {get;set;}
    public Integer exemptAmount {get;set;}
    public String jurisCode {get;set;}
    public String jurisName {get;set;}
    public String stateAssignedNo {get;set;}
    public String jurisType {get;set;}
    public Integer nonTaxableAmount {get;set;}
    public Double rate {get;set;}
    public Double tax {get;set;}
    public Integer taxableAmount {get;set;}
    public String taxType {get;set;}
    public String taxName {get;set;}
    public Integer taxAuthorityTypeId {get;set;}
    public Double taxCalculated {get;set;}
    public String rateType {get;set;}

    public Details(JSONParser parser) {
        while (parser.nextToken() != JSONToken.END_OBJECT) {
```

```
                if (parser.getCurrentToken() == JSONToken.FIELD_NAME) {
                    String text = parser.getText();
                    if (parser.nextToken() != JSONToken.VALUE_NULL) {
                        if (text == 'id') {
                            id = parser.getText();
                        } else if (text == 'transactionLineId') {
                            transactionLineId = parser.getText();
                        } else if (text == 'transactionId') {
                            transactionId = parser.getText();
                        } else if (text == 'country') {
                            country = parser.getText();
                        } else if (text == 'region') {
                            region = parser.getText();
                        } else if (text == 'exemptAmount') {
                            exemptAmount = parser.getIntegerValue();
                        } else if (text == 'jurisCode') {
                            jurisCode = parser.getText();
                        } else if (text == 'jurisName') {
                            jurisName = parser.getText();
                        } else if (text == 'stateAssignedNo') {
                            stateAssignedNo = parser.getText();
                        } else if (text == 'jurisType') {
                            jurisType = parser.getText();
                        } else if (text == 'nonTaxableAmount') {
                            nonTaxableAmount = parser.getIntegerValue();
                        } else if (text == 'rate') {
                            rate = parser.getDoubleValue();
                        } else if (text == 'tax') {
                            tax = parser.getDoubleValue();
                        } else if (text == 'taxableAmount') {
                            taxableAmount = parser.getIntegerValue();
                        } else if (text == 'taxType') {
                            taxType = parser.getText();
                        } else if (text == 'taxName') {
                            taxName = parser.getText();
                        } else if (text == 'taxAuthorityTypeId') {
                            taxAuthorityTypeId = parser.getIntegerValue();
                        } else if (text == 'taxCalculated') {
                            taxCalculated = parser.getDoubleValue();
                        } else if (text == 'rateType') {
                            rateType = parser.getText();
                        } else {
                            consumeObject(parser);
                        }
                    }
                }
            }
        }
    }

    public class Messages {
        public String summary {get;set;}
        public String details {get;set;}
        public String refersTo {get;set;}
```

```
        public String severity {get;set;}
        public String source {get;set;}

        public Messages(JSONParser parser) {
            while (parser.nextToken() != JSONToken.END_OBJECT) {
                if (parser.getCurrentToken() == JSONToken.FIELD_NAME) {
                    String text = parser.getText();
                    if (parser.nextToken() != JSONToken.VALUE_NULL) {
                        if (text == 'summary') {
                            summary = parser.getText();
                        } else if (text == 'details') {
                            details = parser.getText();
                        } else if (text == 'refersTo') {
                            refersTo = parser.getText();
                        } else if (text == 'severity') {
                            severity = parser.getText();
                        } else if (text == 'source') {
                            source = parser.getText();
                        } else {
                            consumeObject(parser);
                        }
                    }
                }
            }
        }
    }

    public String id {get;set;}
    public String code {get;set;}
    public String referenceCode {get;set;}
    public Integer companyId {get;set;}
    public String taxDate {get;set;}
    public String transactionDate {get;set;}
    public String status {get;set;}
    public String type_Z {get;set;} // in json: type
    public Boolean reconciled {get;set;}
    public Integer totalAmount {get;set;}
    public Integer totalExempt {get;set;}
    public Double totalTax {get;set;}
    public Integer totalTaxable {get;set;}
    public Double totalTaxCalculated {get;set;}
    public String adjustmentReason {get;set;}
    public Boolean locked {get;set;}
    public Integer version {get;set;}
    public String modifiedDate {get;set;}
    public Integer modifiedUserId {get;set;}
    public List<Lines> lines {get;set;}
    public List<Addresses> addresses {get;set;}
    public List<Summary> summary {get;set;}
    public List<Messages> messages {get;set;}

    public JsonSuccessParser(JSONParser parser) {
        while (parser.nextToken() != JSONToken.END_OBJECT) {
            if (parser.getCurrentToken() == JSONToken.FIELD_NAME) {
```

```
                String text = parser.getText();
                if (parser.nextToken() != JSONToken.VALUE_NULL) {
                    if (text == 'id') {
                        id = parser.getText();
                    } else if (text == 'code') {
                        code = parser.getText();
                    } else if (text == 'referenceCode'){
                        referenceCode = parser.getText();
                    } else if (text == 'companyId') {
                        companyId = parser.getIntegerValue();
                    } else if (text == 'taxDate') {
                        taxDate = parser.getText();
                    } else if (text == 'date') {
                        transactionDate = parser.getText();
                    } else if (text == 'status') {
                        status = parser.getText();
                    } else if (text == 'type') {
                        type_Z = parser.getText();
                    } else if (text == 'reconciled') {
                        reconciled = parser.getBooleanValue();
                    } else if (text == 'totalAmount') {
                        totalAmount = parser.getIntegerValue();
                    } else if (text == 'totalExempt') {
                        totalExempt = parser.getIntegerValue();
                    } else if (text == 'totalTax') {
                        totalTax = parser.getDoubleValue();
                    } else if (text == 'totalTaxable') {
                        totalTaxable = parser.getIntegerValue();
                    } else if (text == 'totalTaxCalculated') {
                        totalTaxCalculated = parser.getDoubleValue();
                    } else if (text == 'adjustmentReason') {
                        adjustmentReason = parser.getText();
                    } else if (text == 'locked') {
                        locked = parser.getBooleanValue();
                    } else if (text == 'version') {
                        version = parser.getIntegerValue();
                    } else if (text == 'modifiedDate') {
                        modifiedDate = parser.getText();
                    } else if (text == 'modifiedUserId') {
                        modifiedUserId = parser.getIntegerValue();
                    } else if (text == 'lines') {
                        lines = new List<Lines>();
                        while (parser.nextToken() != JSONToken.END_ARRAY) {
                            lines.add(new Lines(parser));
                        }
                    } else if (text == 'addresses') {
                        addresses = new List<Addresses>();
                        while (parser.nextToken() != JSONToken.END_ARRAY) {
                            addresses.add(new Addresses(parser));
                        }
                    } else if (text == 'summary') {
                        summary = new List<Summary>();
                        while (parser.nextToken() != JSONToken.END_ARRAY) {
                            summary.add(new Summary(parser));
```

```
                    }
                } else if (text == 'messages') {
                    messages = new List<Messages>();
                    while (parser.nextToken() != JSONToken.END_ARRAY) {
                        messages.add(new Messages(parser));
                    }
                } else {
                    consumeObject(parser);
                }
            }
        }
    }
}

public class Summary {
    public String country {get;set;}
    public String region {get;set;}
    public String jurisType {get;set;}
    public String jurisCode {get;set;}
    public String jurisName {get;set;}
    public Integer taxAuthorityType {get;set;}
    public String stateAssignedNo {get;set;}
    public String taxType {get;set;}
    public String taxName {get;set;}
    public String taxGroup {get;set;}
    public String rateType {get;set;}
    public Integer taxable {get;set;}
    public Double rate {get;set;}
    public Double tax {get;set;}
    public Double taxCalculated {get;set;}
    public Integer nonTaxable {get;set;}
    public Integer exemption {get;set;}

    public Summary(JSONParser parser) {
        while (parser.nextToken() != JSONToken.END_OBJECT) {
            if (parser.getCurrentToken() == JSONToken.FIELD_NAME) {
                String text = parser.getText();
                if (parser.nextToken() != JSONToken.VALUE_NULL) {
                    if (text == 'country') {
                        country = parser.getText();
                    } else if (text == 'region') {
                        region = parser.getText();
                    } else if (text == 'jurisType') {
                        jurisType = parser.getText();
                    } else if (text == 'jurisCode') {
                        jurisCode = parser.getText();
                    } else if (text == 'jurisName') {
                        jurisName = parser.getText();
                    } else if (text == 'taxAuthorityType') {
                        taxAuthorityType = parser.getIntegerValue();
                    } else if (text == 'stateAssignedNo') {
                        stateAssignedNo = parser.getText();
                    } else if (text == 'taxType') {
                        taxType = parser.getText();
```

```
                        } else if (text == 'taxName') {
                            taxName = parser.getText();
                        } else if (text == 'taxGroup') {
                            taxGroup = parser.getText();
                        } else if (text == 'rateType') {
                            rateType = parser.getText();
                        } else if (text == 'taxable') {
                            taxable = parser.getIntegerValue();
                        } else if (text == 'rate') {
                            rate = parser.getDoubleValue();
                        } else if (text == 'tax') {
                            tax = parser.getDoubleValue();
                        } else if (text == 'taxCalculated') {
                            taxCalculated = parser.getDoubleValue();
                        } else if (text == 'nonTaxable') {
                            nonTaxable = parser.getIntegerValue();
                        } else if (text == 'exemption') {
                            exemption = parser.getIntegerValue();
                        } else {
                            consumeObject(parser);
                        }
                    }
                }
            }
        }
    }

    public class Lines {
        public String id {get;set;}
        public String transactionId {get;set;}
        public String lineNumber {get;set;}
        public Integer discountAmount {get;set;}
        public Integer exemptAmount {get;set;}
        public Integer exemptCertId {get;set;}
        public Boolean isItemTaxable {get;set;}
        public Integer lineAmount {get;set;}
        public Double quantity {get;set;}
        public String reportingDate {get;set;}
        public Double tax {get;set;}
        public Integer taxableAmount {get;set;}
        public Double taxCalculated {get;set;}
        public String taxCode {get;set;}
        public String taxDate {get;set;}
        public Boolean taxIncluded {get;set;}
        public List<Details> details {get;set;}
        public String itemCode {get;set;}
        public Lines(JSONParser parser) {
            while (parser.nextToken() != JSONToken.END_OBJECT) {
                if (parser.getCurrentToken() == JSONToken.FIELD_NAME) {
                    String text = parser.getText();
                    if (parser.nextToken() != JSONToken.VALUE_NULL) {
                        if (text == 'id') {
                            id = parser.getText();
                        } else if (text == 'transactionId') {
```

```
                                transactionId = parser.getText();
                            }else if (text == 'itemCode') {
                                itemCode = parser.getText();
                            }else if (text == 'lineNumber') {
                                lineNumber = parser.getText();
                            } else if (text == 'discountAmount') {
                                discountAmount = parser.getIntegerValue();
                            } else if (text == 'exemptAmount') {
                                exemptAmount = parser.getIntegerValue();
                            } else if (text == 'exemptCertId') {
                                exemptCertId = parser.getIntegerValue();
                            } else if (text == 'isItemTaxable') {
                                isItemTaxable = parser.getBooleanValue();
                            } else if (text == 'lineAmount') {
                                lineAmount = parser.getIntegerValue();
                            } else if (text == 'quantity') {
                                quantity = parser.getDoubleValue();
                            } else if (text == 'reportingDate') {
                                reportingDate = parser.getText();
                            } else if (text == 'tax') {
                                tax = parser.getDoubleValue();
                            } else if (text == 'taxableAmount') {
                                taxableAmount = parser.getIntegerValue();
                            } else if (text == 'taxCalculated') {
                                taxCalculated = parser.getDoubleValue();
                            } else if (text == 'taxCode') {
                                taxCode = parser.getText();
                            } else if (text == 'taxDate') {
                                taxDate = parser.getText();
                            } else if (text == 'taxIncluded') {
                                taxIncluded = parser.getBooleanValue();
                            } else if (text == 'details') {
                                details = new List<Details>();
                                while (parser.nextToken() != JSONToken.END_ARRAY) {
                                    details.add(new Details(parser));
                                }
                            } else {
                                consumeObject(parser);
                            }
                        }
                    }
                }
            }
        }


    public static JsonSuccessParser parse(String json)
    {
        return new JsonSuccessParser(System.JSON.createParser(json));
    }
}
```

Prepare your JSON request to call the Avalara endpoint by using the `AvalaraJSONBuilder` class.

```
public with sharing class AvalaraJSONBuilder
{
    private static AvalaraJSONBuilder avalaraJSONBuilderInstance;

    public static AvalaraJSONBuilder getInstance()
    {
        if (NULL == avalaraJSONBuilderInstance)
        {
            avalaraJSONBuilderInstance = new AvalaraJSONBuilder();
        }
        return avalaraJSONBuilderInstance;
    }

    public String frameJsonForGetTaxOrderItem(commercetax.CalculateTaxRequest
calculateTaxRequest)
    {
        try
        {
            Id accountid  = null;
            if(calculateTaxRequest.CustomerDetails.AccountId != null &&
calculateTaxRequest.CustomerDetails.AccountId != '')
             accountid = Id.valueof(calculateTaxRequest.CustomerDetails.AccountId);

            JSONGenerator jsonGeneratorInstance = JSON.createGenerator(true);
            jsonGeneratorInstance.writeStartObject();
            String type = null;
            if(calculateTaxRequest.taxtype == commercetax.CalculateTaxType.Actual)
                type ='SalesInvoice';
                else type = 'SalesOrder';
            jsonGeneratorInstance.writeStringField('type', type);
            if(calculateTaxRequest.SellerDetails != null)
                jsonGeneratorInstance.writeStringField('companyCode',
calculateTaxRequest.SellerDetails.code);
            else
                jsonGeneratorInstance.writeStringField('companyCode', 'billing2');
            if(calculateTaxRequest.isCommit != null) {
                jsonGeneratorInstance.writeBooleanField('commit',
calculateTaxRequest.isCommit);
            }
            if(calculateTaxRequest.documentcode != null){
                jsonGeneratorInstance.writeStringField('code',
calculateTaxRequest.documentcode);
            }else if(calculateTaxRequest.referenceEntityId != null) {
                jsonGeneratorInstance.writeStringField('code',
calculateTaxRequest.referenceEntityId);
            }
           if(calculateTaxRequest.CustomerDetails.code == null && accountid !=null)
 {
                Account acc = [select id, name from account where id=:accountid];
                jsonGeneratorInstance.writeStringField('customerCode', acc.name);
            } else {
                jsonGeneratorInstance.writeStringField('customerCode',
calculateTaxRequest.CustomerDetails.code);
```

```
            }
            if(calculateTaxRequest.EffectiveDate == null)
                jsonGeneratorInstance.writeDateField('date', system.today());
            else
                jsonGeneratorInstance.writeDateTimeField('date',
calculateTaxRequest.EffectiveDate);

            jsonGeneratorInstance.writeFieldName('lines');
            jsonGeneratorInstance.writeStartArray();
            for(integer i=0;i<1;i++){
                for(Commercetax.TaxLineItemRequest lineItem :
calculateTaxRequest.LineItems)
                {
                    jsonGeneratorInstance.writeStartObject();
                    if(lineItem.linenumber != null){
                        jsonGeneratorInstance.writeStringField('number',
lineItem.linenumber);
                    }
                    jsonGeneratorInstance.writeNumberField('quantity',
lineItem.Quantity);
                    jsonGeneratorInstance.writeNumberField('amount',
(lineItem.Amount));

jsonGeneratorInstance.writeStringField('taxCode',lineItem.taxCode);

                    jsonGeneratorInstance.writeFieldName('addresses');
                    jsonGeneratorInstance.writeStartObject();
                    jsonGeneratorInstance.writeFieldName('ShipFrom');
                    jsonGeneratorInstance.writeStartObject();
                    jsonGeneratorInstance.writeStringField('line1',
lineItem.addresses.shipfrom.street);
                    jsonGeneratorInstance.writeStringField('line2',
lineItem.addresses.shipfrom.street);
                    jsonGeneratorInstance.writeStringField('city',
lineItem.addresses.shipfrom.city);
                    jsonGeneratorInstance.writeStringField('region',
lineItem.addresses.shipfrom.state);
                    jsonGeneratorInstance.writeStringField('country',
lineItem.addresses.shipfrom.country);

jsonGeneratorInstance.writeStringField('postalCode',lineItem.addresses.shipfrom.postalcode);

                    jsonGeneratorInstance.writeEndObject();

                    jsonGeneratorInstance.writeFieldName('ShipTo');
                    jsonGeneratorInstance.writeStartObject();
                    jsonGeneratorInstance.writeStringField('line1',
lineItem.addresses.shipto.street);
                    jsonGeneratorInstance.writeStringField('line2',
lineItem.addresses.shipto.street);
                    jsonGeneratorInstance.writeStringField('city',
lineItem.addresses.shipto.city);
                    jsonGeneratorInstance.writeStringField('region',
lineItem.addresses.shipto.state);
```

```
                    jsonGeneratorInstance.writeStringField('country',
lineItem.addresses.shipto.country);

jsonGeneratorInstance.writeStringField('postalCode',lineItem.addresses.shipto.postalcode);

                    jsonGeneratorInstance.writeEndObject();

                    jsonGeneratorInstance.writeFieldName('pointOfOrderOrigin');
                    jsonGeneratorInstance.writeStartObject();
                    jsonGeneratorInstance.writeStringField('line1',
lineItem.addresses.soldto.street);
                    jsonGeneratorInstance.writeStringField('line2',
lineItem.addresses.soldto.street);
                    jsonGeneratorInstance.writeStringField('city',
lineItem.addresses.soldto.city);
                    jsonGeneratorInstance.writeStringField('region',
lineItem.addresses.soldto.state);
                    jsonGeneratorInstance.writeStringField('country',
lineItem.addresses.soldto.country);

jsonGeneratorInstance.writeStringField('postalCode',lineItem.addresses.soldto.postalcode);

                    jsonGeneratorInstance.writeEndObject();


                    if(lineItem.effectiveDate != null)
                    {
                        jsonGeneratorInstance.writeFieldName('taxOverride');
                        jsonGeneratorInstance.writeStartObject();
                        jsonGeneratorInstance.writeDateTimeField('taxDate',
lineItem.effectiveDate);
                        jsonGeneratorInstance.writeEndObject();
                    }
                    jsonGeneratorInstance.writeEndObject();
                    jsonGeneratorInstance.writeEndObject();
                }
            }
                jsonGeneratorInstance.writeEndArray();
            jsonGeneratorInstance.writeEndObject();
            return jsonGeneratorInstance.getAsString();
        }
        catch (Exception e)
        {
             throw e;
        }
    }
}
```

- Use the `JsonErrorParser` class to extract the error details, if any.

```
global with sharing class JsonErrorParser
{
    public cls_error error;
```

```
    public class cls_error
    {
        public String code;
        public String message;
        public String target;
        public cls_details[] details;
    }

    public class cls_details
    {
        public String code;
        public String message;
        public String description;
        public String faultCode;
        public String helpLink;
        public String severity;
    }
    public static JsonErrorParser parse(String json)
    {
       return (JsonErrorParser) System.JSON.deserialize(json, JsonErrorParser.class);

    }
}
```

TaxEngineContext Class

Wrapper class that stores details about the type of a tax calculation request.

Namespace

CommerceTax

Example

At the beginning of a tax adapter, use `TaxEngineContext` class to pass the value of a request type to an instance of `RequestType`.

```
global virtual class MockAdapter implements commercetax.TaxEngineAdapter {

     global commercetax.TaxEngineResponse processRequest(commercetax.TaxEngineContext
taxEngineContext) {
         commercetax.RequestType requestType = taxEngineContext.getRequestType();
         commercetax.CalculateTaxRequest request =
(commercetax.CalculateTaxRequest)taxEngineContext.getRequest();
```

Build the rest of your adapter based on the type of request that you got from `TaxEngineContext` class.

```
if(requestType == commercetax.RequestType.CalculateTax){
             commercetax.calculatetaxtype type = request.taxtype;
             String docCode='';
             if(request.DocumentCode == 'simulateEmptyDocumentCode')
                 docCode = '';
             else if(request.DocumentCode != null)
                 docCode =request.DocumentCode;
```

```
            else if(request.ReferenceEntityId != null) docCode = request.ReferenceEntityId;

              else docCode =  String.valueOf(getRandomInteger(0,2147483647));
              commercetax.CalculateTaxResponse response = new
commercetax.CalculateTaxResponse();
              if(request.isCommit == true) {
                  response.setStatus(commercetax.TaxTransactionStatus.Committed);
              } else {
                  response.setStatus(commercetax.TaxTransactionStatus.Uncommitted);
              }
}
```

TaxEngineContext Constructors

Learn more about the available constructors with the `TaxEngineContext` class.

TaxEngineContext Methods

Learn more about the available methods with the `TaxEngineContext` class.

TaxEngineContext Constructors

Learn more about the available constructors with the `TaxEngineContext` class.

The `TaxEngineContext` class includes these constructors.

TaxEngineContext(request, requestType, namedUri)

Initializes the `TaxEngineContext` object. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

`TaxEngineContext(request, requestType, namedUri)`

Initializes the `TaxEngineContext` object. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

Signature

```
TaxEngineContext(commercetax.TaxEngineRequest request, commercetax.RequestType
requestType, String namedUri)
```

Parameters

`request`

Type: TaxEngineRequest

Information about the request.

`requestType`

Type: RequestType

Whether the tax request is to calculate or estimate tax.

`namedUri`

Type: String

URI that was called as part of the tax calculation request.

TaxEngineContext Methods

Learn more about the available methods with the `TaxEngineContext` class.

The `TaxEngineContext` class includes these methods.

getNamedUri()
Retrieves the value of the NamedUri field of the `TaxEngineContext` class.

getRequest()
Gets the value of the `TaxEngineContext`'s Request field.

getRequestType()
Gets the value of the RequestType field of the `TaxEngineContext` class.

`getNamedUri()`

Retrieves the value of the NamedUri field of the `TaxEngineContext` class.

Signature

```
global String getNamedUri()
```

Return Value

Type: String

`getRequest()`

Gets the value of the `TaxEngineContext`'s Request field.

Signature

```
global commercetax.TaxEngineRequest getRequest()
```

Return Value

Type: TaxEngineRequest

An implemented instance of an external tax engine's interface for processing requests. We've provided the `TaxEngineRequest` interface for you to test within mock adapters with classes that implement it, such as CalculateTaxRequest. However, don't use it outside of a testing context.

`getRequestType()`

Gets the value of the RequestType field of the `TaxEngineContext` class.

Signature

```
global commercetax.RequestType getRequestType()
```

Return Value

Type: RequestType

Indicates whether the calculation request was for actual or calculated tax.

TaxLineItemRequest Class

Contains line item details of a tax request.

Namespace

CommerceTax

TaxLineItemRequest Constructors
Learn more about the constructors available with the `TaxLineItemRequest` class.

TaxLineItemRequest Properties
Learn more about the available properties with the `TaxLineItemRequest` class.

TaxLineItemRequest Methods
Learn more about the available methods with the `TaxLineItemRequest` class.

TaxLineItemRequest Constructors

Learn more about the constructors available with the `TaxLineItemRequest` class.

The `TaxLineItemRequest` class includes these constructors.

TaxLineItemRequest(addresses, amount, description, productCode, quantity, lineNumber, taxCode, effectiveDate)
Initializes the request for the tax line item. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

`TaxLineItemRequest(addresses, amount, description, productCode, quantity, lineNumber, taxCode, effectiveDate)`

Initializes the request for the tax line item. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

Signature

```
global TaxLineItemRequest(commercetax.LineTaxAddressesRequest addresses, Double amount,
String description, String productCode, Double quantity, String lineNumber, String
taxCode, Datetime effectiveDate)
```

Parameters

addresses
Type: LineTaxAddressesRequest

Information about the addresses applied to each line item in a tax calculation request.

amount
Type: Double

Total amount (in a given currency) represented by a line item sent for tax calculation.

description
Type: String

User-defined description for a tax line item.

productCode
Type: String

Catalog code for the product represented by the tax line item.

quantity
Type: Double

The number of units of a given product that the tax line item represents.

lineNumber
Type: String

Unique number used to identify a tax line item.

taxCode
Type: String

Code used to identify how tax is calculated for a tax line item.

effectiveDate
Type: Datetime

This is a user-defined date used for reporting only.

TaxLineItemRequest Properties

Learn more about the available properties with the `TaxLineItemRequest` class.

The `TaxLineItemRequest` class includes these properties.

addresses
Contains the list of addresses of a line item.

amount
Total amount (in a given currency) represented by a line item sent for tax calculation.

description
User-defined description for a tax line item.

effectiveDate
The date that a tax transaction takes effect on a line item. This is a user-defined date used for reporting only.

lineNumber
Unique number used to identify a tax line item.

productCode
Catalog code for the product represented by the tax line item.

quantity
Number of units of a given product that the tax line item represents.

taxCode
Code used to identify how tax is calculated for a tax line item.

addresses

Contains the list of addresses of a line item.

Signature

```
public commercetax.LineTaxAddressesRequest addresses {get; set;}
```

Property Value

Type: commercetax.LineTaxAddressesRequest

amount

Total amount (in a given currency) represented by a line item sent for tax calculation.

Signature

```
global Double amount {get; set;}
```

Property Value

Type: Double

description

User-defined description for a tax line item.

Signature

```
global String description {get; set;}
```

Property Value

Type: String

effectiveDate

The date that a tax transaction takes effect on a line item. This is a user-defined date used for reporting only.

Signature

```
global Datetime effectiveDate {get; set;}
```

Property Value

Type: Datetime

`lineNumber`

Unique number used to identify a tax line item.

Signature

```
global String lineNumber {get; set;}
```

Property Value

Type: String

`productCode`

Catalog code for the product represented by the tax line item.

Signature

```
global String productCode {get; set;}
```

Property Value

Type: String

`quantity`

Number of units of a given product that the tax line item represents.

Signature

```
global Double quantity {get; set;}
```

Property Value

Type: Double

`taxCode`

Code used to identify how tax is calculated for a tax line item.

Signature

```
global String taxCode {get; set;}
```

Property Value

Type: String

TaxLineItemRequest Methods

Learn more about the available methods with the `TaxLineItemRequest` class.

The `TaxLineItemRequest` class includes these methods.

equals(obj)

Maintains the integrity of lists of type `TaxLineItemRequest` 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 `TaxLineItemRequest` by determining the uniqueness of the external object records in a list.

toString()

Converts a value to a string.

`equals(obj)`

Maintains the integrity of lists of type `TaxLineItemRequest` 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 `TaxLineItemRequest` by determining the uniqueness of the external object records in a list.

Signature

```
global Integer hashCode()
```

Return Value

Type: Integer

`toString()`

Converts a value to a string.

Signature

```
global String toString()
```

Return Value

Type: String

TaxSellerDetailsRequest Class

Contains tax code details used in the tax calculation request.

Namespace

CommerceTax

TaxSellerDetailsRequest Constructors
Learn more about the available constructors with the `TaxSellerDetailsRequest` class.

TaxSellerDetailsRequest Properties
Learn more about the available properties with the `TaxSellerDetailsRequest` class.

TaxSellerDetailsRequest Methods
Learn more about the available methods with the `TaxSellerDetailsRequest` class.

TaxSellerDetailsRequest Constructors

Learn more about the available constructors with the `TaxSellerDetailsRequest` class.

The `TaxSellerDetailsRequest` class includes these constructors.

TaxSellerDetailsRequest(code)
Initializes the request for the tax seller details. This constructor is intended for test usage and throws an exception if used outside of the Apex test context

`TaxSellerDetailsRequest(code)`

Initializes the request for the tax seller details. This constructor is intended for test usage and throws an exception if used outside of the Apex test context

Signature

```
global TaxSellerDetailsRequest(String code)
```

Parameters

code
Type: String

Tax code used for tax calculation.

TaxSellerDetailsRequest Properties

Learn more about the available properties with the `TaxSellerDetailsRequest` class.

The `TaxSellerDetailsRequest` class includes these properties.

code
Tax code used for tax calculation.

code

Tax code used for tax calculation.

Signature

```
global String code {get; set;}
```

Property Value

Type: String

TaxSellerDetailsRequest Methods

Learn more about the available methods with the `TaxSellerDetailsRequest` class.

The `TaxSellerDetailsRequest` class includes these methods.

equals(obj)
Maintains the integrity of lists of type `TaxSellerDetailsRequest` by determining the equality of the 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 `TaxSellerDetailsRequest` by determining the uniqueness of the external objects in a list.

toString()
Converts a value to a string.

equals(obj)

Maintains the integrity of lists of type `TaxSellerDetailsRequest` by determining the equality of the 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 `TaxSellerDetailsRequest` by determining the uniqueness of the external objects in a list.

Signature

```
global Integer hashCode()
```

Return Value

Type: Integer

`toString()`

Converts a value to a string.

Signature

```
global String toString()
```

Return Value

Type: String

TaxTransactionRequest Class

Abstract class for storing customer details used in tax calculation and estimation requests.

Namespace

CommerceTax

Usage

Specify the `CommerceTax` namespace when creating an instance of this class. The constructor of this class takes no arguments. For example, let's say you create an instance of `CalculateTaxRequest` class, which extends the `TaxTransactionRequest` class.

TaxTransactionRequest Constructors
Learn more about the available constructors with the `TaxTransactionRequest` class.

TaxTransactionRequest Properties
Learn more about the available properties with the `TaxTransactionRequest` class.

TaxTransactionRequest Methods

TaxTransactionRequest Constructors

Learn more about the available constructors with the `TaxTransactionRequest` class.

The `TaxTransactionRequest` class includes these constructors.

TaxTransactionRequest(addresses, currencyIsoCode, customerDetails, description, documentCode, referenceDocumentCode, transactionDate, effectiveDate, lineItems, referenceEntityId, sellerDetails)

Initializes the request for the tax transaction. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

TaxTransactionRequest(addresses, currencyIsoCode, customerDetails, description, documentCode, referenceDocumentCode, transactionDate, effectiveDate, lineItems, referenceEntityId, sellerDetails)

Initializes the request for the tax transaction. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

Signature

```
global TaxTransactionRequest(commercetax.HeaderTaxAddressesRequest addresses, String
currencyIsoCode, commercetax.TaxCustomerDetailsRequest customerDetails, String
description, String documentCode, String referenceDocumentCode, Datetime transactionDate,
Datetime effectiveDate, List<commercetax.TaxLineItemRequest> lineItems, String
referenceEntityId, commercetax.TaxSellerDetailsRequest sellerDetails)
```

Parameters

addresses

Type: HeaderTaxAddressesRequest

Tax addresses, such as Ship To and Bill From.

currencyIsoCode

Type: String

Three-letter ISO 4217 currency code associated with the `TaxTransactionRequest`.

customerDetails

Type: TaxCustomerDetailsRequest

Customer information used in tax calculation.

description

Type: String

Optional user-defined description for providing more information about the tax transaction request.

documentCode

Type: String

Code for documents that are used to provide more information in the tax calculation process.

referenceDocumentCode

Type: String

Code for additional documents that are used in the tax calculation process.

transactionDate

Type: Datetime

The date that the tax transaction occurred.

effectiveDate
Type: Datetime

The date that the tax transaction takes effect. User-defined and used only for reporting purposes.

lineItems
Type: List<TaxLineItemRequest>

A list of line items on which tax is calculated.

referenceEntityId
Type: String

ID of an object related to the line items sent for tax calculation.

sellerDetails
Type: TaxSellerDetailsRequest

Contains tax code information used in a tax calculation request.

TaxTransactionRequest Properties

Learn more about the available properties with the `TaxTransactionRequest` class.

The `TaxTransactionRequest` class includes these properties.

addresses
A list of addresses (such as Ship To and Sold To) used as part of the tax transaction.

currencyIsoCode
Three-letter ISO 4217 currency code associated with the `TaxTransactionRequest`.

customerDetails
Customer information used in tax calculation.

description
Optional user-defined description for providing more information about the tax transaction request.

documentCode
Code for documents used to provide more information in the tax calculation process.

effectiveDate
The date that the tax transaction takes effect. User-defined and used only for reporting purposes.

lineItems
A list of line items on which tax will be calculated.

referenceDocumentCode
Code for documents used to provide more information in the tax calculation process.

referenceEntityId
ID of an object related to the line items sent for tax calculation.

sellerDetails
Contains tax code information used in a tax calculation request.

transactionDate
The date that the tax transaction occurred.

addresses

A list of addresses (such as Ship To and Sold To) used as part of the tax transaction.

Signature

```
global commercetax.HeaderTaxAddressesRequest addresses {get; set;}
```

Property Value

Type: HeaderTaxAddressesRequest

currencyIsoCode

Three-letter ISO 4217 currency code associated with the `TaxTransactionRequest`.

Signature

```
global String currencyIsoCode {get; set;}
```

Property Value

Type: String

customerDetails

Customer information used in tax calculation.

Signature

```
global CommerceTax.TaxCustomerDetailsRequest customerDetails {get; set;}
```

Property Value

Type: TaxCustomerDetailsRequest

description

Optional user-defined description for providing more information about the tax transaction request.

Signature

```
global String description {get; set;}
```

Property Value

Type: String

documentCode

Code for documents used to provide more information in the tax calculation process.

Signature

```
global String documentCode {get; set;}
```

Property Value

Type: String

effectiveDate

The date that the tax transaction takes effect. User-defined and used only for reporting purposes.

Signature

```
global Datetime effectiveDate {get; set;}
```

Property Value

Type: Datetime

lineItems

A list of line items on which tax will be calculated.

Signature

```
global List<CommerceTax.TaxLineItemRequest> lineItems {get; set;}
```

Property Value

Type: List<TaxLineItemRequest>

referenceDocumentCode

Code for documents used to provide more information in the tax calculation process.

Signature

```
global String referenceDocumentCode {get; set;}
```

Property Value

Type: String

referenceEntityId

ID of an object related to the line items sent for tax calculation.

Signature

```
global String referenceEntityId {get; set;}
```

Property Value

Type: String

sellerDetails

Contains tax code information used in a tax calculation request.

Signature

```
global commercetax.TaxSellerDetailsRequest sellerDetails {get; set;}
```

Property Value

Type: TaxSellerDetailsRequest

transactionDate

The date that the tax transaction occurred.

Signature

```
global Datetime transactionDate {get; set;}
```

Property Value

Type: Datetime

TaxTransactionRequest Methods

The following are methods for `TaxTransactionRequest`.

equals(obj)
Maintains the integrity of lists of type `TaxTransactionRequest` 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 `TaxTransactionRequest` by determining the uniqueness of the external object records in a list.

toString()
Converts a value to a string.

equals(obj)

Maintains the integrity of lists of type `TaxTransactionRequest` 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

Return Value

Type: Boolean

hashCode()

Maintains the integrity of lists of type `TaxTransactionRequest` by determining the uniqueness of the external object records in a list.

Signature

```
global Integer hashCode()
```

Return Value

Type: Integer

toString()

Converts a value to a string.

Signature

```
global String toString()
```

Return Value

Type: String

TaxTransactionStatus Enum

Shows whether the tax transaction has been committed or uncommitted.

Usage

Used by the CalculateTaxResponse class method.

Enum Values

The `commercetax.TaxTransactionStatus` enum includes these values.

Value	Description
`Committed`	Tax has been calculated and committed.
`Uncommitted`	Tax has been calculated but hasn't been committed.

TaxTransactionType Enum

Shows whether the tax transaction is for a credit or debit transaction.

Usage

Used by the CalculateTaxResponse and CalculateTaxRequest class methods.

Enum Values

The `commercetax.TaxTransactionType` enum includes these values.

Value	Description
`Credit`	Represents a credit transaction.
`Debit`	Represents a debit transaction.

PlaceQuote Namespace

The `PlaceQuote` namespace provides classes and methods to create or update quotes with pricing preferences and configuration options.

The `PlaceQuote` namespace includes these classes.

CatalogRatesPreferenceEnum Enum

Specifies the rate card entries defined in the catalog that must be fetched for quote line items, with usage-based selling during the quote creation process.

ConfigurationInputEnum Enum

Specifies the configuration input for the request to place a quote.

ConfigurationOptionsInput Class

Contains methods and properties to set the configuration options for the input to the product configurator.

GraphRequest Class

Contains constructors and properties to set the graph ID and a list of records to be ingested. The list of records is specified in a key-value map format that contains the field values of an order.

PlaceQuoteException Class

Contains methods to hold the exception details for the place quote request.

PlaceQuoteResponse Class

Contains properties to hold the response to the place quote request.

PlaceQuoteRLMApexProcessor Class

Contains methods to place a quote with details of the graph request, pricing preferences, and configuration options.

PricingPreferenceEnum Enum

Specifies the pricing preference during the create quote process.

RecordResource Class

Contains constructors and properties to create a record object from the field values of a quote.

RecordWithReferenceRequest Class

Contains constructors and properties to associate a record object with a reference identifier.

CatalogRatesPreferenceEnum Enum

Specifies the rate card entries defined in the catalog that must be fetched for quote line items, with usage-based selling during the quote creation process.

Usage

This enum is available when the Usage-Based Selling feature is enabled.

Enum Values

The `placequote.CatalogRatesPreferenceEnum` enum includes these values.

Value	Description
`Fetch`	Retrieves the rate card entries defined in the catalog for quote line items during the quote creation process.
`Skip`	Skips the retrieval of rate card entries for quote line items during the quote creation process. The default value is `Skip`.

ConfigurationInputEnum Enum

Specifies the configuration input for the request to place a quote.

Usage

Use these enum values for the `configurationInputEnum` property in the PlaceQuoteRLMApexProcessor class.

Enum Values

The `placequote.ConfigurationInputEnum` enum has these values.

Value	Description
`RunAndAllowErrors`	Run the configuration and proceed with order ingestion upon encountering any configuration errors.
`RunAndBlockErrors`	Run the configuration and block order ingestion upon encountering any configuration errors.
`Skip`	Skip the configuration execution.

ConfigurationOptionsInput Class

Contains methods and properties to set the configuration options for the input to the product configurator.

Namespace

PlaceQuote

Usage

This class holds the required details of the product configuration input. Set the class properties to enable default configuration, execution of configuration rules, and validation of the product catalog. Use these class properties as an input to the PlaceQuoteRLMApexProcessor class method.

Example

```
PlaceQuote.GraphRequest graph = new PlaceQuote.GraphRequest('test',listOfRecords);
    PlaceQuote.PricingPreferenceEnum pricingPreference =
PlaceQuote.PricingPreferenceEnum.System;
    PlaceQuote.ConfigurationInputEnum configurationPreference =
PlaceQuote.ConfigurationInputEnum.RunAndAllowErrors;
   PlaceQuote.ConfigurationOptionsInput cInput = new PlaceQuote.ConfigurationOptionsInput();

    cInput.addDefaultConfiguration = true;
    cInput.executeConfigurationRules = true;
    cInput.validateAmendRenewCancel = true;
    cInput.validateProductCatalog = true;
        //Place Quote Call
     PlaceQuote.PlaceQuoteResponse resp =
PlaceQuote.PlaceQuoteRLMApexProcessor.execute(pricingPreference,graph,configurationPreference,cInput);
```

ConfigurationOptionsInput Properties

Set the `ConfigurationOptionsInput` class properties to add default configuration, execute configuration rules, and validate the product catalog.

ConfigurationOptionsInput Methods

Learn more about the methods available with the `ConfigurationOptionsInput` class.

ConfigurationOptionsInput Properties

Set the `ConfigurationOptionsInput` class properties to add default configuration, execute configuration rules, and validate the product catalog.

The `ConfigurationOptionsInput` class includes these properties.

addDefaultConfiguration

Sets the default product configuration, such as bundle and product attributes, for a quote request.

executeConfigurationRules

Sets the requirement for a quote to adhere to the configuration rules.

validateAmendRenewCancel

Sets the requirement to run validations related to amend, renew, or cancel processes.

validateProductCatalog

Sets the requirement to validate a quote against the product catalog.

`addDefaultConfiguration`

Sets the default product configuration, such as bundle and product attributes, for a quote request.

Signature

```
public Boolean addDefaultConfiguration {get; set;}
```

Property Value

Type: Boolean

Indicates whether to automatically add default configuration to the order (`true`) or not (`false`).

`executeConfigurationRules`

Sets the requirement for a quote to adhere to the configuration rules.

Signature

```
public Boolean executeConfigurationRules {get; set;}
```

Property Value

Type: Boolean

Indicates whether the order must adhere to configuration rules during processing (`true`) or bypass them (`false`).

`validateAmendRenewCancel`

Sets the requirement to run validations related to amend, renew, or cancel processes.

Signature

```
public Boolean validateAmendRenewCancel {get; set;}
```

Property Value

Type: Boolean

Indicates whether to run validations related to amend, renew, or cancel processes (`true`) or not (`false`).

`validateProductCatalog`

Sets the requirement to validate a quote against the product catalog.

Signature

```
public Boolean validateProductCatalog {get; set;}
```

Property Value

Type: Boolean

Indicates whether the quote must be validated against the product catalog (`true`) or not (`false`).

ConfigurationOptionsInput Methods

Learn more about the methods available with the `ConfigurationOptionsInput` class.

The `ConfigurationOptionsInput` class includes these methods.

equals(obj)
Determines the equality of external objects in a list. This method is dynamic and is based on the `equals()` method in Java.

hashCode()
Determines the uniqueness of the external object records in a list.

toString()
Converts a value to a string.

`equals(obj)`

Determines the equality of external objects in a list. This method is dynamic and is based on the `equals()` method in Java.

Signature

```
public Boolean equals(Object obj)
```

Parameters

obj
Type: Object

Reference object that's used to compare with the class object.

Return Value

Type: Boolean

Indicates if the class object is same as the reference object (`true`) or not (`false`).

`hashCode()`

Determines the uniqueness of the external object records in a list.

Signature

```
public Integer hashCode()
```

Return Value

Type: Integer

Integer hash code that represents the value of the object. Equal objects as per the `equals()` method must return the same hash code.

`toString()`

Converts a value to a string.

Signature

```
public String toString()
```

Return Value

Type: String

GraphRequest Class

Contains constructors and properties to set the graph ID and a list of records to be ingested. The list of records is specified in a key-value map format that contains the field values of an order.

Namespace

PlaceQuote

Example

Invoke the Place Quote Apex API by using these steps.

- Set up a quote and quote line item. To associate the Record object with a reference identifier, create an instance of the `RecordWithReferenceRequest` class.

```
        //Quote Setup
    PlaceQuote.RecordResource quoteRecord = new
PlaceQuote.RecordResource(Quote.getSobjectType(),'POST');
    Map<String,Object> quoteFieldValues = new Map<String,Object>();
    quoteFieldValues.put('Name','q-ap12');
    quoteFieldValues.put('OpportunityId','006xx000001aBFcAAM');
    quoteFieldValues.put('Pricebook2Id','01sxx0000005pvRAAQ');
    quoteRecord.fieldValues = quoteFieldValues;

           //Quote Line Item Setup
    PlaceQuote.RecordResource quoteLineItemRecord1 = new
PlaceQuote.RecordResource(QuoteLineItem.getSobjectType(),'POST');
    Map<String,Object> quoteLineItemFieldValues = new Map<String,Object>();
    quoteLineItemFieldValues.put('Product2Id','01txx0000006ibwAAA');
    quoteLineItemFieldValues.put('PricebookEntryId','01uxx0000008zPqAAI');
    quoteLineItemFieldValues.put('Quantity','2.0');
    quoteLineItemFieldValues.put('UnitPrice','5.0');
    quoteLineItemFieldValues.put('StartDate','2023-03-15');
    quoteLineItemFieldValues.put('QuoteId','@{refQuote.id}');
    quoteLineItemRecord1.fieldValues = quoteLineItemFieldValues;
    PlaceQuote.RecordWithReferenceRequest quoteItemRecords = new
PlaceQuote.RecordWithReferenceRequest('refQuote',quoteRecord);
    PlaceQuote.RecordWithReferenceRequest quoteLineItemRecords1 = new
PlaceQuote.RecordWithReferenceRequest('refQuoteItem1',quoteLineItemRecord1);
```

- To create a quote line relationship, create an instance of the `RecordResource` class.

```
PlaceQuote.RecordResource quoteLineRelationship1 = new
PlaceQuote.RecordResource(QuoteLineRelationship.getSobjectType(),'POST');
    Map<String,Object> quoteLineRelationshipValues = new Map<String,Object>();
    quoteLineRelationshipValues.put('ProductRelationshipTypeId','0yoxx00000000JNAAY');
    quoteLineRelationshipValues.put('MainQuoteLineId','@{refQuoteItem2.id}');
    quoteLineRelationshipValues.put('AssociatedQuoteLineId','@{refQuoteItem1.id}');
   quoteLineRelationshipValues.put('AssociatedQuoteLinePricing','IncludedInBundlePrice');

    quoteLineRelationship1.fieldValues = quoteLineRelationshipValues;
```

- Create the list of records to be ingested.

```
    PlaceQuote.RecordWithReferenceRequest quoteLineRelationship = new
PlaceQuote.RecordWithReferenceRequest('QuoteLineRelationship',quoteLineRelationship1);
    List<PlaceQuote.RecordWithReferenceRequest> listOfRecords = new
List<PlaceQuote.RecordWithReferenceRequest>();
    listOfRecords.add(quoteItemRecords);
    listOfRecords.add(quoteLineItemRecords1);
    listOfRecords.add(quoteLineItemRecords2);
    listOfRecords.add(quoteLineRelationship);
```

- To contain all record objects, create an instance of the `GraphRequest` class.

  Note: The `CatalogRatesPreferenceEnum` enum is available when the Usage-Based Selling feature is enabled.

```
    PlaceQuote.GraphRequest graph = new PlaceQuote.GraphRequest('test',listOfRecords);
    PlaceQuote.ConfigurationOptionsInput cInput = new
PlaceQuote.ConfigurationOptionsInput();
    PlaceQuote.CatalogRatesPreferenceEnum catalogRatesPreference =
PlaceQuote.CatalogRatesPreferenceEnum.Fetch;
....PlaceQuote.ConfigurationInputEnum configurationPreference =
PlaceQuote.ConfigurationInputEnum.RunAndAllowErrors;
    PlaceQuote.PricingPreferenceEnum pricingPreference =
PlaceQuote.PricingPreferenceEnum.System;
    //System.debug(graph);

    //Place Quote Call
    PlaceQuote.PlaceQuoteResponse resp =
PlaceQuote.PlaceQuoteRLMApexProcessor.execute(pricingPreference, catalogRatesPreference,
 graph, configurationPreference, cInput);
    System.debug(resp);
```

GraphRequest Constructors

Learn more about the available constructors with the `GraphRequest` class.

GraphRequest Properties

GraphRequest Constructors

Learn more about the available constructors with the `GraphRequest` class.

The `GraphRequest` class includes these constructors.

GraphRequest(graphId, records)
Creates an instance of the `GraphRequest` class to assign the graph ID and a list of records to be ingested.

`GraphRequest(graphId, records)`

Creates an instance of the `GraphRequest` class to assign the graph ID and a list of records to be ingested.

Signature

```
public GraphRequest(String graphId, List<placequote.RecordWithReferenceRequest> records)
```

Parameters

`graphId`
Type: String

ID of the graph.

`records`
Type: List<placequote.RecordWithReferenceRequest on page 903 >

List of records to be ingested.

GraphRequest Properties

The following are properties for `GraphRequest`.

graphId
Set the `graphId` property to assign the ID value of the graph.

`graphId`

Set the `graphId` property to assign the ID value of the graph.

Signature

```
public String graphId {get; set;}
```

Property Value

Type: String

PlaceQuoteException Class

Contains methods to hold the exception details for the place quote request.

Namespace

PlaceQuote

PlaceQuoteException Methods
Learn more about the methods available with the `PlaceQuoteException` class.

PlaceQuoteException Methods

Learn more about the methods available with the `PlaceQuoteException` class.

The `PlaceQuoteException` class includes these methods.

getErrorCode()
Gets the error code that's associated to the place quote request.

`getErrorCode()`

Gets the error code that's associated to the place quote request.

Signature

```
public String getErrorCode()
```

Return Value

Type: String

PlaceQuoteResponse Class

Contains properties to hold the response to the place quote request.

Namespace

PlaceQuote

Example

```
PlaceQuote.PlaceQuoteResponse resp =
PlaceQuote.PlaceQuoteExecutor.execute(internalEnum,graph);
```

PlaceQuoteResponse Properties
Learn more about the available properties with the `PlaceQuoteResponse` class.

PlaceQuoteResponse Properties

Learn more about the available properties with the `PlaceQuoteResponse` class.

The `PlaceQuoteResponse` class includes these properties.

quoteId
Get the ID of the quote that's created after a successful request.

requestIdentifier
Get the request ID of the process to query the asynchronous status of the Place Quote Apex API.

responseError
Get the list of errors encountered during the synchronous processing of the API request.

statusURL
Get the asynchronous status URL of the request, if available.

success
Get the request status of the synchronous part of the processing.

quoteId

Get the ID of the quote that's created after a successful request.

Signature

```
public String quoteId {get; set;}
```

Property Value

Type: String

requestIdentifier

Get the request ID of the process to query the asynchronous status of the Place Quote Apex API.

Signature

```
public String requestIdentifier {get; set;}
```

Property Value

Type: String

responseError

Get the list of errors encountered during the synchronous processing of the API request.

Signature

```
public List<ConnectApi.PlaceQuoteErrorResponse> responseError {get; set;}
```

Property Value

Type: List<ConnectApi.PlaceQuoteErrorResponse>

statusURL

Get the asynchronous status URL of the request, if available.

Signature

```
public String statusURL {get; set;}
```

Property Value

Type: String

`success`

Get the request status of the synchronous part of the processing.

Signature

```
public Boolean success {get; set;}
```

Property Value

Type: Boolean

Indicates whether the synchronous part of the processing is successful (`true`) or not (`false`).

PlaceQuoteRLMApexProcessor Class

Contains methods to place a quote with details of the graph request, pricing preferences, and configuration options.

Namespace

PlaceQuote

PlaceQuoteRLMApexProcessor Methods
Learn more about the methods available with the `PlaceQuoteRLMApexProcessor` class.

PlaceQuoteRLMApexProcessor Example Implementation
To place a quote from Apex, refer to the example implementation of the `PlaceQuoteRLMApexProcessor` class.

PlaceQuoteRLMApexProcessor Methods

Learn more about the methods available with the `PlaceQuoteRLMApexProcessor` class.

The `PlaceQuoteRLMApexProcessor` class includes these methods.

execute(pricingPreferenceEnum, graphRequest, configurationInputEnum, configurationOptionsInput)
Use the method in the `PlaceQuoteExecutor` class to execute the Place Quote Apex API request by assigning the properties for graph request, pricing references, and configuration options.

execute(pricingPreferenceEnum, catalogRatesPreference, graphRequest, configurationInputEnum, configurationOptionsInput)
Use the method in the `PlaceQuoteExecutor` class to execute the Place Quote Apex API request by assigning the properties for graph request, pricing references, and configuration options. This method also includes the property to define fetching of rate card entries.

`execute(pricingPreferenceEnum, graphRequest, configurationInputEnum, configurationOptionsInput)`

Use the method in the `PlaceQuoteExecutor` class to execute the Place Quote Apex API request by assigning the properties for graph request, pricing references, and configuration options.

Signature

```
public static placequote.PlaceQuoteResponse execute(placequote.PricingPreferenceEnum
pricingPreferenceEnum, placequote.GraphRequest graphRequest,
placequote.ConfigurationInputEnum configurationInputEnum,
placequote.ConfigurationOptionsInput configurationOptionsInput)
```

Parameters

`pricingPreferenceEnum`
Type: placequote.PricingPreferenceEnum

Pricing preference during the quote process.

`graphRequest`
Type: placequote.GraphRequest

The sObject graph values of the quote payload to be ingested.

`configurationInputEnum`
Type: placequote.ConfigurationInputEnum

Configuration input for the quote process.

`configurationOptionsInput`
Type: placequote.ConfigurationOptionsInput

Configuration options during the ingestion process.

Return Value

Type: placequote.PlaceQuoteResponse

`execute(pricingPreferenceEnum, catalogRatesPreference, graphRequest, configurationInputEnum, configurationOptionsInput)`

Use the method in the `PlaceQuoteExecutor` class to execute the Place Quote Apex API request by assigning the properties for graph request, pricing references, and configuration options. This method also includes the property to define fetching of rate card entries.

Signature

```
public static placequote.PlaceQuoteResponse execute(placequote.PricingPreferenceEnum
pricingPreferenceEnum, placequote.CatalogRatesPreferenceEnum catalogRatesPreference,
placequote.GraphRequest graphRequest, placequote.ConfigurationInputEnum
configurationInputEnum, placequote.ConfigurationOptionsInput configurationOptionsInput)
```

Parameters

pricingPreferenceEnum
Type: placequote.PricingPreferenceEnum

Pricing preference during the quote process.

catalogRatesPreference
Type: placequote.CatalogRatesPreferenceEnum

The rate card entries defined in the catalog that must be fetched for quote line items, with usage-based pricing during the quote creation process. The `CatalogRatesPreferenceEnum` enum is available when the Usage-Based Selling feature is enabled.

graphRequest
Type: placequote.GraphRequest

The sObject graph values of the quote payload to be ingested.

configurationInputEnum
Type: placequote.ConfigurationInputEnum

Configuration input for the quote process.

configurationOptionsInput
Type: placequote.ConfigurationOptionsInput

Configuration options during the ingestion process.

Return Value

Type: placequote.PlaceQuoteResponse

PlaceQuoteRLMApexProcessor Example Implementation

To place a quote from Apex, refer to the example implementation of the `PlaceQuoteRLMApexProcessor` class.

Namespace

PlaceQuote

Usage

Customize this example to suit your requirements. Create the list of records to be ingested by using these steps. Replace the respective IDs with the values that are present in your org. For example, replace the value of `${Pricebook2Id}` field with the price book ID that's present in the org.

Example:

- Set up a quote and quote line item. To associate the Record object with a reference identifier, create an instance of the `RecordWithReferenceRequest` class.

```
        //Quote Setup
    PlaceQuote.RecordResource quoteRecord = new
PlaceQuote.RecordResource(Quote.getSobjectType(),'POST');
    Map<String,Object> quoteFieldValues = new Map<String,Object>();
    quoteFieldValues.put('Name','q-ap12');
    quoteFieldValues.put('OpportunityId','006xx000001aBFcAAM');
```

```
    quoteFieldValues.put('Pricebook2Id','01sxx0000005pvRAAQ');
    quoteRecord.fieldValues = quoteFieldValues;

           //Quote Line Item Setup
    PlaceQuote.RecordResource quoteLineItemRecord1 = new
PlaceQuote.RecordResource(QuoteLineItem.getSobjectType(),'POST');
    Map<String,Object> quoteLineItemFieldValues = new Map<String,Object>();
    quoteLineItemFieldValues.put('Product2Id','01txx0000006ibwAAA');
    quoteLineItemFieldValues.put('PricebookEntryId','01uxx0000008zPqAAI');
    quoteLineItemFieldValues.put('Quantity','2.0');
    quoteLineItemFieldValues.put('UnitPrice','5.0');
    quoteLineItemFieldValues.put('StartDate','2023-03-15');
    quoteLineItemFieldValues.put('QuoteId','@{refQuote.id}');
    quoteLineItemRecord1.fieldValues = quoteLineItemFieldValues;
    PlaceQuote.RecordWithReferenceRequest quoteItemRecords = new
PlaceQuote.RecordWithReferenceRequest('refQuote',quoteRecord);
    PlaceQuote.RecordWithReferenceRequest quoteLineItemRecords1 = new
PlaceQuote.RecordWithReferenceRequest('refQuoteItem1',quoteLineItemRecord1);
```

- To create a quote line relationship, create an instance of the `RecordResource` class.

```
PlaceQuote.RecordResource quoteLineRelationship1 = new
PlaceQuote.RecordResource(QuoteLineRelationship.getSobjectType(),'POST');
    Map<String,Object> quoteLineRelationshipValues = new Map<String,Object>();
   quoteLineRelationshipValues.put('ProductRelationshipTypeId','0yoxx00000000JNAAY');

    quoteLineRelationshipValues.put('MainQuoteLineId','@{refQuoteItem2.id}');
   quoteLineRelationshipValues.put('AssociatedQuoteLineId','@{refQuoteItem1.id}');


quoteLineRelationshipValues.put('AssociatedQuoteLinePricing','IncludedInBundlePrice');

    quoteLineRelationship1.fieldValues = quoteLineRelationshipValues;
```

- Create the list of records to be ingested.

```
    PlaceQuote.RecordWithReferenceRequest quoteLineRelationship = new
PlaceQuote.RecordWithReferenceRequest('QuoteLineRelationship',quoteLineRelationship1);

    List<PlaceQuote.RecordWithReferenceRequest> listOfRecords = new
List<PlaceQuote.RecordWithReferenceRequest>();
    listOfRecords.add(quoteItemRecords);
    listOfRecords.add(quoteLineItemRecords1);
    listOfRecords.add(quoteLineItemRecords2);
    listOfRecords.add(quoteLineRelationship);
```

- To contain all record objects, create an instance of the `GraphRequest` class.

Note: The `CatalogRatesPreferenceEnum` enum is available when the Usage-Based Selling feature is enabled.

```
   PlaceQuote.GraphRequest graph = new PlaceQuote.GraphRequest('test',listOfRecords);
```

```
    PlaceQuote.ConfigurationOptionsInput cInput = new
PlaceQuote.ConfigurationOptionsInput();
    PlaceQuote.CatalogRatesPreferenceEnum catalogRatesPreference =
PlaceQuote.CatalogRatesPreferenceEnum.Fetch;
....PlaceQuote.ConfigurationInputEnum configurationPreference =
PlaceQuote.ConfigurationInputEnum.RunAndAllowErrors;
    PlaceQuote.PricingPreferenceEnum pricingPreference =
PlaceQuote.PricingPreferenceEnum.System;
    //System.debug(graph);

    //Place Quote Call
    PlaceQuote.PlaceQuoteResponse resp =
PlaceQuote.PlaceQuoteRLMApexProcessor.execute(pricingPreference,
catalogRatesPreference, graph, configurationPreference, cInput);
    System.debug(resp);
```

PricingPreferenceEnum Enum

Specifies the pricing preference during the create quote process.

Usage

Used by the PlaceQuoteRLMApexProcessor class.

Enum Values

The `placequote.PricingPreferenceEnum` enum class includes these values.

Value	Description
`Force`	Enforce pricing during the quote ingestion process.
`Skip`	Skip pricing during the quote ingestion process.
`System`	Determine whether a pricing calculation is required.

RecordResource Class

Contains constructors and properties to create a record object from the field values of a quote.

Namespace

PlaceQuote

Example

```
PlaceQuote.RecordResource quoteLineRelationship1 = new
PlaceQuote.RecordResource(QuoteLineRelationship.getSobjectType(),'POST');
```

See PlaceQuoteRLMApexProcessor to refer to an example implementation.

RecordResource Constructors

Learn more about the available constructors with the `RecordResource` class.

RecordResource Properties

Learn more about the available properties with the `RecordResource` class.

RecordResource Constructors

Learn more about the available constructors with the `RecordResource` class.

The `RecordResource` class includes these constructors.

RecordResource(type, method, id)

Creates an instance of the `RecordResource` class to assign values to the fields of a quote item by using the sObject type, API method, and quote ID properties.

RecordResource(type, method)

Creates an instance of the `RecordResource` class to assign the values to the fields of a quote item by using the sObject type and API method properties.

`RecordResource(type, method, id)`

Creates an instance of the `RecordResource` class to assign values to the fields of a quote item by using the sObject type, API method, and quote ID properties.

Signature

```
public RecordResource(Schema.SObjectType type, String method, Id id)
```

Parameters

type

Type: Schema.SObjectType

Object that's returned from the field describe result using the `getReferenceTo()` method or from the sObject describe result using the `getSObjectType()` method.

method

Type: String

Method for the API request, such as POST or PATCH.

id

Type: Id

ID of the quote.

`RecordResource(type, method)`

Creates an instance of the `RecordResource` class to assign the values to the fields of a quote item by using the sObject type and API method properties.

Signature

```
public RecordResource(Schema.SObjectType type, String method)
```

Parameters

type
Type: Schema.SObjectType

Object that's returned from the field describe result using the `getReferenceTo()` method or from the sObject describe result using the `getSObjectType()` method.

method
Type: String

Method for the API request, such as POST or PATCH.

RecordResource Properties

Learn more about the available properties with the `RecordResource` class.

The `RecordResource` class includes these properties.

fieldValues
Set the `fieldValues` property to assign values to the fields to update the quote record.

id
Set the `id` property to assign the ID of the quote record.

method
Set the `method` property to specify the API request method, such as POST or PATCH.

type
Set the type property to assign the object type that's returned from the field describe result by using the `getReferenceTo()` method or from the sObject describe result by using the `getSObjectType()` method.

`fieldValues`

Set the `fieldValues` property to assign values to the fields to update the quote record.

Signature

```
public Map<String,ANY> fieldValues {get; set;}
```

Property Value

Type: List <Map<String,ANY>>

`id`

Set the `id` property to assign the ID of the quote record.

Signature

```
public String id {get; set;}
```

Property Value

Type: String

`method`

Set the `method` property to specify the API request method, such as POST or PATCH.

Signature

```
public String method {get; set;}
```

Property Value

Type: String

`type`

Set the type property to assign the object type that's returned from the field describe result by using the `getReferenceTo()` method or from the sObject describe result by using the `getSObjectType()` method.

Signature

```
public Schema.SObjectType type {get; set;}
```

Property Value

Type: Schema.SObjectType

RecordWithReferenceRequest Class

Contains constructors and properties to associate a record object with a reference identifier.

Namespace

PlaceQuote

Example

```
PlaceQuote.RecordWithReferenceRequest quoteLineRelationship = new
PlaceQuote.RecordWithReferenceRequest('QuoteLineRelationship',quoteLineRelationship1);
```

See PlaceQuoteRLMApexProcessor to refer to an example implementation.

RecordWithReferenceRequest Constructors

Learn more about the available constructors with the `RecordWithReferenceRequest` class.

RecordWithReferenceRequest Properties

Learn more about the available properties with the `RecordWithReferenceRequest` class.

RecordWithReferenceRequest Constructors

Learn more about the available constructors with the `RecordWithReferenceRequest` class.

The `RecordWithReferenceRequest` class includes these constructors.

RecordWithReferenceRequest(referenceId, record)

Creates an instance of the `RecordWithReferenceRequest` class to associate a record object with a reference identifier by using the `referenceId` and `record` object properties.

`RecordWithReferenceRequest(referenceId, record)`

Creates an instance of the `RecordWithReferenceRequest` class to associate a record object with a reference identifier by using the `referenceId` and `record` object properties.

Signature

```
public RecordWithReferenceRequest(String referenceId, placequote.RecordResource record)
```

Parameters

`referenceId`

Type: String

Reference ID that maps to the subrequest response and can be used to reference the response in subsequent subrequests. You can reference the referenceId in either the body or URL of a subrequest. Use this syntax to include a reference: @{referenceId.FieldName}. See referenceId property of a composite subrequest.

`record`

Type: placequote.RecordResource

Record object that's defined using the `RecordResource` class.

RecordWithReferenceRequest Properties

Learn more about the available properties with the `RecordWithReferenceRequest` class.

The `RecordWithReferenceRequest` class includes these properties.

record

Set the `record` property to specify the record object that's defined by using the `RecordResource` class.

referenceId

Set the `referenceId` property to specify the reference ID that maps to the subrequest response. This reference ID can be used to reference the response in subsequent subrequests.

`record`

Set the `record` property to specify the record object that's defined by using the `RecordResource` class.

Signature

```
public placequote.RecordResource record {get; set;}
```

Property Value

Type: placequote.RecordResource

`referenceId`

Set the `referenceId` property to specify the reference ID that maps to the subrequest response. This reference ID can be used to reference the response in subsequent subrequests.

Signature

```
public String referenceId {get; set;}
```

Property Value

Type: String

RevSalesTrxn Namespace

Create a sales transaction, such as a quote or an order, with integrated pricing and configuration. Additionally, update an order or a quote, and insert and delete order or quote line items to calculate the estimated tax.

The `RevSalesTrxn` namespace includes these classes.

CatalogRatesPreferenceEnum Enum
Specifies the rate card entries defined in the catalog that must be fetched for quote line items, with usage-based selling during the place sales transaction process.

ConfigurationExecutionEnum Enum
Specifies the configuration method for the place sales transaction request.

ConfigurationOptionsInput Class
Contains methods and properties to set the configuration options for the input to the product configurator.

GraphRequest Class
Contains constructors and properties to set the graph ID and a list of records to be ingested. The list of records is specified in a key-value map format that contains the field values of an order.

PlaceSalesTransactionException Class
Contains methods to hold the exception details for the place sales transaction request.

PlaceSalesTransactionExecutor Class
Contains methods to place a sales transaction with details of the graph request, pricing preferences, and configuration options.

PlaceSalesTransactionResponse Class
Contains properties to hold the response to the place sales transaction request.

PricingPreferenceEnum Enum
Specifies the pricing preference during the creation of a sales transaction.

RecordResource Class
Contains constructors and properties to create a record object from the field values of a sales transaction.

RecordWithReferenceRequest Class
Contains constructors and properties to associate a record object with a reference identifier.

CatalogRatesPreferenceEnum Enum

Specifies the rate card entries defined in the catalog that must be fetched for quote line items, with usage-based selling during the place sales transaction process.

Usage

This enum is available when the Usage-Based Selling feature is enabled.

Enum Values

The `RevSalesTrxn.CatalogRatesPreferenceEnum` enum includes these values.

Value	Description
`Fetch`	Retrieves the rate card entries defined in the catalog for quote line items during the quote creation process.
`Skip`	Skips the retrieval of rate card entries for quote line items during the quote creation process. The default value is `Skip`.

ConfigurationExecutionEnum Enum

Specifies the configuration method for the place sales transaction request.

Usage

Use these enum values for the `configurationExecutionEnum` property in the PlaceSalesTransactionExecutor class.

Enum Values

The `RevSalesTrxn.ConfigurationExecutionEnum` enum has these values.

Value	Description
`Force`	Specifies to enforce the predefined configuration process during the sales transaction process.
`Skip`	Specifies to skip the configuration process during the quote creation process. The default value is `Skip`.
`System`	Specifies the system to determine whether the configuration process is required.

ConfigurationOptionsInput Class

Contains methods and properties to set the configuration options for the input to the product configurator.

Namespace

RevSalesTrxn

Usage

This class holds the required details of the product configuration input. Set the class properties to enable default configuration, execution of configuration rules, and validation of the product catalog. Use these class properties as an input to the PlaceSalesTransactionExecutor class method.

Example

```
        RevSalesTrxn.GraphRequest graph = new RevSalesTrxn.GraphRequest('test',
listOfRecords);
    RevSalesTrxn.PricingPreferenceEnum pricingPrefEnum =
RevSalesTrxn.PricingPreferenceEnum.SYSTEM;
    RevSalesTrxn.ConfigurationExecutionEnum configurationExecutionEnum =
RevSalesTrxn.ConfigurationExecutionEnum.SYSTEM;
    RevSalesTrxn.ConfigurationOptionsInput cInput = new
RevSalesTrxn.ConfigurationOptionsInput();
    cInput.addDefaultConfiguration = true;
    cInput.executeConfigurationRules = true;
    cInput.validateAmendRenewCancel = true;
    cInput.validateProductCatalog = true;
        //Place Sales Transaction API Call
     RevSalesTrxn.PlaceSalesTransactionResponse resp =
PlaceQuote.PlaceSalesTransactionExecutor.execute(graph,pricingPrefEnum,configurationExecutionEnum,cInput,null);
```

ConfigurationOptionsInput Properties

Set the ConfigurationOptionsInput class properties to add default configuration, execute configuration rules, and validate the product catalog.

ConfigurationOptionsInput Methods

Learn more about the methods available with the ConfigurationOptionsInput class.

ConfigurationOptionsInput Properties

Set the ConfigurationOptionsInput class properties to add default configuration, execute configuration rules, and validate the product catalog.

The `ConfigurationOptionsInput` class includes these properties.

addDefaultConfiguration

Sets the default product configuration, such as bundle and product attributes, for a quote request.

executeConfigurationRules

Sets the requirement for a quote to adhere to the configuration rules.

validateAmendRenewCancel

Sets the requirement to run validations related to amend, renew, or cancel processes.

validateProductCatalog

Sets the requirement to validate a quote against the product catalog.

`addDefaultConfiguration`

Sets the default product configuration, such as bundle and product attributes, for a quote request.

Signature

```
public Boolean addDefaultConfiguration {get; set;}
```

Property Value

Type: Boolean

Indicates whether to automatically add default configuration to the order (`true`) or not (`false`).

`executeConfigurationRules`

Sets the requirement for a quote to adhere to the configuration rules.

Signature

```
public Boolean executeConfigurationRules {get; set;}
```

Property Value

Type: Boolean

Indicates whether the order must adhere to configuration rules during processing (`true`) or bypass them (`false`).

`validateAmendRenewCancel`

Sets the requirement to run validations related to amend, renew, or cancel processes.

Signature

```
public Boolean validateAmendRenewCancel {get; set;}
```

Property Value

Type: Boolean

Indicates whether to run validations related to amend, renew, or cancel processes (`true`) or not (`false`).

`validateProductCatalog`

Sets the requirement to validate a quote against the product catalog.

Signature

```
public Boolean validateProductCatalog {get; set;}
```

Property Value

Type: Boolean

Indicates whether the quote must be validated against the product catalog (`true`) or not (`false`).

ConfigurationOptionsInput Methods

Learn more about the methods available with the ConfigurationOptionsInput class.

The `ConfigurationOptionsInput` class includes these methods.

equals(obj)
Determines the equality of external objects in a list. This method is dynamic and is based on the equals() method in Java.

hashCode()
Determines the uniqueness of the external object records in a list.

toString()
Converts a value to a string.

`equals(obj)`

Determines the equality of external objects in a list. This method is dynamic and is based on the equals() method in Java.

Signature

```
public Boolean equals(Object obj)
```

Parameters

obj
Type: Object

Reference object that's used to compare with the class object.

Return Value

Type: Boolean

Indicates if the class object is same as the reference object (`true`) or not (`false`).

`hashCode()`

Determines the uniqueness of the external object records in a list.

Signature

```
public Integer hashCode()
```

Return Value

Type: Integer

Integer hash code that represents the value of the object. Equal objects as per the `equals()` method must return the same hash code.

`toString()`

Converts a value to a string.

Signature

```
public String toString()
```

Return Value

Type: String

GraphRequest Class

Contains constructors and properties to set the graph ID and a list of records to be ingested. The list of records is specified in a key-value map format that contains the field values of an order.

Namespace

RevSalesTrxn

GraphRequest Constructors
Learn more about the available constructors with the GraphRequest class.

GraphRequest Properties
Learn more about the available properties with the GraphRequest class.

GraphRequest Constructors

Learn more about the available constructors with the GraphRequest class.

The `GraphRequest` class includes these constructors.

GraphRequest(graphId, records)
Creates an instance of the GraphRequest class to assign the graph ID and a list of records to be ingested.

`GraphRequest(graphId, records)`

Creates an instance of the GraphRequest class to assign the graph ID and a list of records to be ingested.

Signature

```
public GraphRequest(String graphId, List<RevSalesTrxn.RecordWithReferenceRequest>
records)
```

Parameters

`graphId`
Type: String

ID of the graph.

`records`
Type: List<placequote.RecordWithReferenceRequest on page 923>

List of records to be ingested.

GraphRequest Properties

Learn more about the available properties with the GraphRequest class.

The `GraphRequest` class includes these properties.

graphId
Set the `graphId` property to assign the ID value of the graph.

`graphId`

Set the `graphId` property to assign the ID value of the graph.

Signature

```
public String graphId {get; set;}
```

Property Value

Type: String

PlaceSalesTransactionException Class

Contains methods to hold the exception details for the place sales transaction request.

Namespace

RevSalesTrxn

PlaceSalesTransactionException Methods
Learn more about the methods available with the PlaceSalesTransactionException class.

PlaceSalesTransactionException Methods

Learn more about the methods available with the PlaceSalesTransactionException class.

The `PlaceSalesTransactionException` class includes these methods.

getErrorCode()
Gets the error code that's associated to the place sales transaction request.

`getErrorCode()`

Gets the error code that's associated to the place sales transaction request.

Signature

```
public String getErrorCode()
```

Return Value

Type: String

PlaceSalesTransactionExecutor Class

Contains methods to place a sales transaction with details of the graph request, pricing preferences, and configuration options.

Namespace

RevSalesTrxn

PlaceSalesTransactionExecutor Methods

Learn more about the methods available with the `PlaceSalesTransactionExecutor` class.

PlaceSalesTransactionExecutor Example Implementation

To place a sales transaction from Apex, refer to the example implementation of the `PlaceSalesTransactionExecutor` class.

PlaceSalesTransactionExecutor Methods

Learn more about the methods available with the `PlaceSalesTransactionExecutor` class.

The `PlaceSalesTransactionExecutor` class includes these methods.

execute(graphRequest, pricingPreferenceEnum, configurationExecutionEnum, configuratorOptions, id)

Use the method in the `PlaceSalesTransactionExecutor` class to execute the Place Sales Transaction Apex API request by assigning the properties for graph request, pricing references, and configurator options.

`execute(graphRequest, pricingPreferenceEnum, configurationExecutionEnum, configuratorOptions, id)`

Use the method in the `PlaceSalesTransactionExecutor` class to execute the Place Sales Transaction Apex API request by assigning the properties for graph request, pricing references, and configurator options.

Signature

```
public static revsalestrxn.PlaceSalesTransactionResponse
execute(revsalestrxn.GraphRequest graphRequest, revsalestrxn.PricingPreferenceEnum
pricingPreferenceEnum, revsalestrxn.ConfigurationExecutionEnum
configurationExecutionEnum, revsalestrxn.ConfiguratorOptions configuratorOptions,
revsalestrxn.Id id)
```

Parameters

`graphRequest`

Type: revsalestrxn.GraphRequest

The sObject graph values of the order payload to be ingested.

`pricingPreferenceEnum`

Type: revsalestrxn.PricingPreferenceEnum

Pricing preference during the sales transaction process.

configurationExecutionEnum
Type: revsalestrxn.ConfigurationExecutionEnum

Configuration method for the sales transaction request.

configuratorOptions
Type: revsalestrxn.ConfigurationOptionsInput

Configuration options during the creation of the sales transaction.

id
Type: String

ID to assign to the sales transaction.

Return Value

Type: revsalestrxn.PlaceSalesTransactionResponse

PlaceSalesTransactionExecutor Example Implementation

To place a sales transaction from Apex, refer to the example implementation of the `PlaceSalesTransactionExecutor` class.

Namespace

RevSalesTrxn

Usage

Customize this example to suit your requirements. Create the list of records to be ingested by using these steps. Replace the respective IDs with the values that are present in your org. For example, replace the value of `${Pricebook2Id}` field with the price book ID that's present in the org.

Example:

- Set up a quote and quote line item. To associate the Record object with a reference identifier, create an instance of the `RecordWithReferenceRequest` class.
- To create a quote line relationship, create an instance of the `RecordResource` class.
- Create the list of records to be ingested.
- To contain all record objects, create an instance of the `GraphRequest` class.

This example shows a sample request to create a sales transaction, to update a quote with a quote line item, or to delete a quote line item.

```
public class PlaceSalesTransactionTest {
    public static void callPSTAPI_Post() {
        RevSalesTrxn.PricingPreferenceEnum pricingPrefEnum =
RevSalesTrxn.PricingPreferenceEnum.SYSTEM;
        RevSalesTrxn.ConfigurationExecutionEnum configurationExecutionEnum =
RevSalesTrxn.ConfigurationExecutionEnum.SYSTEM;

        //Quote setup
        RevSalesTrxn.RecordResource quoteRecord = new
RevSalesTrxn.RecordResource(Quote.getSobjectType(),'POST');
```

```
        Map<String,Object> quoteFieldValues = new Map<String,Object>();
        quoteFieldValues.put('Name','q-ap12');
        quoteFieldValues.put('OpportunityId','006xx000001a3e8AAA');
        quoteFieldValues.put('Pricebook2Id','01sDU000000JRX8YAO');
        quoteRecord.fieldValues = quoteFieldValues;

        //Quote line item setup

        //1st quote line item
        RevSalesTrxn.RecordResource quoteLineItemRecord1 = new
RevSalesTrxn.RecordResource(QuoteLineItem.getSobjectType(),'POST');
        Map<String,Object> quoteLineItemFieldValues = new Map<String,Object>();
        quoteLineItemFieldValues.put('Product2Id','01txx0000006i7JAAQ');
        quoteLineItemFieldValues.put('PricebookEntryId','01uxx0000008yc6AAA');
        quoteLineItemFieldValues.put('Quantity','2.0');
        quoteLineItemFieldValues.put('UnitPrice','1000');
        quoteLineItemFieldValues.put('StartDate','2025-03-15');
        quoteLineItemFieldValues.put('QuoteId','@{refQuote.id}');
        quoteLineItemRecord1.fieldValues = quoteLineItemFieldValues;
        RevSalesTrxn.RecordWithReferenceRequest quoteItemRecords = new
RevSalesTrxn.RecordWithReferenceRequest('refQuote',quoteRecord);
        RevSalesTrxn.RecordWithReferenceRequest quoteLineItemRecords1 = new
RevSalesTrxn.RecordWithReferenceRequest('refQuoteItem1',quoteLineItemRecord1);

        //2nd quote line item
        RevSalesTrxn.RecordResource quoteLineItemRecord2 = new
RevSalesTrxn.RecordResource(QuoteLineItem.getSobjectType(),'POST');
        Map<String,Object> quoteLineItemFieldValues2 = new Map<String,Object>();
        quoteLineItemFieldValues2.put('Product2Id','01txx0000006i7RAAQ');
        quoteLineItemFieldValues2.put('PricebookEntryId','01uxx0000008ybvAAA');
        quoteLineItemFieldValues2.put('Quantity','2.0');
        quoteLineItemFieldValues2.put('UnitPrice','7.0');
        quoteLineItemFieldValues2.put('StartDate','2025-03-15');
        quoteLineItemFieldValues2.put('QuoteId','@{refQuote.id}');
        quoteLineItemRecord2.fieldValues = quoteLineItemFieldValues2;
        RevSalesTrxn.RecordWithReferenceRequest quoteLineItemRecords2 = new
RevSalesTrxn.RecordWithReferenceRequest('refQuoteItem2',quoteLineItemRecord2);

        List<RevSalesTrxn.RecordWithReferenceRequest> listOfRecords = new
List<RevSalesTrxn.RecordWithReferenceRequest>();
        listOfRecords.add(quoteItemRecords);
        listOfRecords.add(quoteLineItemRecords1);
        listOfRecords.add(quoteLineItemRecords2);


        RevSalesTrxn.GraphRequest graph = new
RevSalesTrxn.GraphRequest('test',listOfRecords);
        System.debug(graph);

        //Place sales transaction API call
        RevSalesTrxn.PlaceSalesTransactionResponse resp =
RevSalesTrxn.PlaceSalesTransactionExecutor.execute(graph, pricingPrefEnum,
configurationExecutionEnum, new RevSalesTrxn.ConfigurationOptionsInput(), null);
        System.debug(resp);
```

```
    }

    public static void callPSTAPI_Patch() {

        // Apex test to update a quote with a quote line item
        RevSalesTrxn.PricingPreferenceEnum pricingPrefEnum =
RevSalesTrxn.PricingPreferenceEnum.SYSTEM;
        RevSalesTrxn.ConfigurationExecutionEnum configurationExecutionEnum =
RevSalesTrxn.ConfigurationExecutionEnum.SYSTEM;

        //Quote setup
        RevSalesTrxn.RecordResource quoteRecord = new
RevSalesTrxn.RecordResource(Quote.getSobjectType(),'PATCH','0Q0xx0000004CYsCAM');
        RevSalesTrxn.RecordWithReferenceRequest quoteItemRecords = new
RevSalesTrxn.RecordWithReferenceRequest('refQuote',quoteRecord);

        //Quote line item setup
        //New quote line item
        RevSalesTrxn.RecordResource quoteLineItemRecord = new
RevSalesTrxn.RecordResource(QuoteLineItem.getSobjectType(),'POST');
        Map<String,Object> quoteLineItemFieldValues = new Map<String,Object>();
        quoteLineItemFieldValues.put('Product2Id','01txx0000006i7KAAQ');
        quoteLineItemFieldValues.put('PricebookEntryId','01uxx0000008ycFAAQ');
        quoteLineItemFieldValues.put('Quantity','2.0');
        quoteLineItemFieldValues.put('UnitPrice','7.0');
        quoteLineItemFieldValues.put('StartDate','2025-03-15');
        quoteLineItemFieldValues.put('QuoteId','@{refQuote.id}');
        quoteLineItemRecord.fieldValues = quoteLineItemFieldValues;
        RevSalesTrxn.RecordWithReferenceRequest quoteLineItemRecords = new
RevSalesTrxn.RecordWithReferenceRequest('refQuoteItem',quoteLineItemRecord);
        List<RevSalesTrxn.RecordWithReferenceRequest> listOfRecords = new
List<RevSalesTrxn.RecordWithReferenceRequest>();
        listOfRecords.add(quoteItemRecords);
        listOfRecords.add(quoteLineItemRecords);
        RevSalesTrxn.GraphRequest graph = new RevSalesTrxn.GraphRequest('test',
listOfRecords);

        //Place sales transaction API call
        RevSalesTrxn.PlaceSalesTransactionResponse resp =
RevSalesTrxn.PlaceSalesTransactionExecutor.execute(graph, pricingPrefEnum,
configurationExecutionEnum, new RevSalesTrxn.ConfigurationOptionsInput(), null);
        System.debug(resp);
    }

    public static void callPSTAPI_Delete() {

        // Apex test to update a quote with a quote line item
        RevSalesTrxn.PricingPreferenceEnum pricingPrefEnum =
RevSalesTrxn.PricingPreferenceEnum.SYSTEM;
        RevSalesTrxn.ConfigurationExecutionEnum configurationExecutionEnum =
RevSalesTrxn.ConfigurationExecutionEnum.SYSTEM;

        //Quote setup
        RevSalesTrxn.RecordResource quoteRecord = new
```

```
RevSalesTrxn.RecordResource(Quote.getSobjectType(),'PATCH','0Q0xx0000004CYsCAM');
        RevSalesTrxn.RecordWithReferenceRequest quoteItemRecords = new
RevSalesTrxn.RecordWithReferenceRequest('refQuote',quoteRecord);

        //Quote line item setup
        //Delete a quote line item
        RevSalesTrxn.RecordResource quoteLineItemRecord = new
RevSalesTrxn.RecordResource(QuoteLineItem.getSobjectType(),'DELETE','0QLxx0000004CYsCAM');

        RevSalesTrxn.RecordWithReferenceRequest quoteLineItemRecords = new
RevSalesTrxn.RecordWithReferenceRequest('refQuoteItem',quoteLineItemRecord);
        List<RevSalesTrxn.RecordWithReferenceRequest> listOfRecords = new
List<RevSalesTrxn.RecordWithReferenceRequest>();
        listOfRecords.add(quoteItemRecords);
        listOfRecords.add(quoteLineItemRecords);
        RevSalesTrxn.GraphRequest graph = new
RevSalesTrxn.GraphRequest('test',listOfRecords);

        //Place sales transaction API call
        RevSalesTrxn.PlaceSalesTransactionResponse resp =
        RevSalesTrxn.PlaceSalesTransactionExecutor.execute(graph, pricingPrefEnum,
configurationExecutionEnum, new RevSalesTrxn.ConfigurationOptionsInput(), null);
        System.debug(resp);
    }
}
```

PlaceSalesTransactionResponse Class

Contains properties to hold the response to the place sales transaction request.

Namespace

RevSalesTrxn

Example

```
RevSalesTrxn.PlaceSalesTransactionResponse resp =
RevSalesTrxn.PlaceSalesTransactionExecutor.execute(graph, pricingPrefEnum,
configurationExecutionEnum, new RevSalesTrxn.ConfigurationOptionsInput(), null);
```

PlaceSalesTransactionResponse Properties

Learn more about the available properties with the PlaceSalesTransactionResponse class.

PlaceSalesTransactionResponse Properties

Learn more about the available properties with the PlaceSalesTransactionResponse class.

The `PlaceSalesTransactionResponse` class includes these properties.

contextDetails

Get the details of the context that's created for the sales transaction.

errorResponse
Get the list of errors encountered during the synchronous processing of the API request.

isSuccess
Get the request status of the synchronous part of the processing.

salesTransactionId
Get the ID of the sales transaction, such as a quote or an order.

statusUrl
Get the asynchronous status URL of the request, if available.

trackerId
Get the unique identifier assigned to a specific operation or request that's used for tracking and referencing the operation.

`contextDetails`

Get the details of the context that's created for the sales transaction.

Signature

```
public ConnectApi.ContextDetails contextDetails {get; set;}
```

Property Value

Type: ConnectApi.ContextDetails

`errorResponse`

Get the list of errors encountered during the synchronous processing of the API request.

Signature

```
public List<ConnectApi.PlaceSalesTransactionErrorResponse> errorResponse {get; set;}
```

Property Value

Type: List<ConnectApi.PlaceSalesTransactionErrorResponse>

`isSuccess`

Get the request status of the synchronous part of the processing.

Signature

```
public Boolean isSuccess {get; set;}
```

Property Value

Type: Boolean

Indicates whether the synchronous part of the processing is successful (`true`) or not (`false`).

`salesTransactionId`

Get the ID of the sales transaction, such as a quote or an order.

Signature

```
public String salesTransactionId {get; set;}
```

Property Value

Type: String

`statusUrl`

Get the asynchronous status URL of the request, if available.

Signature

```
public String statusUrl {get; set;}
```

Property Value

Type: String

`trackerId`

Get the unique identifier assigned to a specific operation or request that's used for tracking and referencing the operation.

Signature

```
public String trackerId {get; set;}
```

Property Value

Type: String

PricingPreferenceEnum Enum

Specifies the pricing preference during the creation of a sales transaction.

Usage

Used by the PlaceSalesTransactionExecutor class.

Enum Values

The `RevSalesTrxn.PricingPreferenceEnum` enum includes these values.

Value	Description
`Force`	Specifies to enforce pricing during the creation of sales transactions.

Value	Description
`Skip`	Specifies to skip pricing during the creation of sales transactions.
`System`	Specifies the system to determine whether a pricing calculation is required. The default value is `System`.

RecordResource Class

Contains constructors and properties to create a record object from the field values of a sales transaction.

Namespace

RevSalesTrxn

Example

```
RevSalesTrxn.RecordResource quoteRecord = new
RevSalesTrxn.RecordResource(Quote.getSobjectType(),'POST');
```

RecordResource Constructors

Learn more about the available constructors with the RecordResource class.

RecordResource Properties

Learn more about the available properties with the RecordResource class.

RecordResource Constructors

Learn more about the available constructors with the RecordResource class.

The `RecordResource` class has these constructors.

RecordResource(type, method, groupAction, criteria)

Creates an instance of the RecordResource class to assign values to the fields of a sales transaction by using the sObject type, API method, and sales transaction ID properties. Additionally, you can group order or quote line items based on a criteria by using the groupAction and criteria properties.

RecordResource(type, method, id)

Creates an instance of the RecordResource class to assign values to the fields of a sales transaction by using the sObject type, API method, and sales transaction ID properties.

RecordResource(type, method)

Creates an instance of the RecordResource class to assign the values to the fields of a sales transaction by using the sObject type and API method properties.

`RecordResource(type, method, groupAction, criteria)`

Creates an instance of the RecordResource class to assign values to the fields of a sales transaction by using the sObject type, API method, and sales transaction ID properties. Additionally, you can group order or quote line items based on a criteria by using the groupAction and criteria properties.

Signature

```
public RecordResource(Schema.SObjectType type, String method, String groupAction,
Map<String,ANY> criteria)
```

Parameters

type
Type: Schema.SObjectType

Object that's returned from the field describe result using the `getReferenceTo()` method or from the sObject describe result using the `getSObjectType()` method.

method
Type: String

Method for the API request, such as POST, PATCH, or DELETE.

groupAction
Type: String

Action to group order or quote line items. Valid values are:

- `GroupBy`
- `Group`
- `Ungroup`
- `GroupAll`
- `DeleteGroup`

criteria
Type: Map<String,ANY>

Criteria to group order or quote line items. For example, group order or quote line items based on a monthly billing frequency.

`RecordResource(type, method, id)`

Creates an instance of the RecordResource class to assign values to the fields of a sales transaction by using the sObject type, API method, and sales transaction ID properties.

Signature

```
public RecordResource(Schema.SObjectType type, String method, Id id)
```

Parameters

type
Type: Schema.SObjectType

Object that's returned from the field describe result using the `getReferenceTo()` method or from the sObject describe result using the `getSObjectType()` method.

method
Type: String

Method for the API request, such as POST, PATCH, or DELETE.

id

Type: Id

ID of the sales transaction, such as a quote or an order.

`RecordResource(type, method)`

Creates an instance of the RecordResource class to assign the values to the fields of a sales transaction by using the sObject type and API method properties.

Signature

```
public RecordResource(Schema.SObjectType type, String method)
```

Parameters

type

Type: Schema.SObjectType

Object that's returned from the field describe result using the `getReferenceTo()` method or from the sObject describe result using the `getSObjectType()` method.

method

Type: String

Method for the API request, such as POST, PATCH, or DELETE.

RecordResource Properties

Learn more about the available properties with the RecordResource class.

The `RecordResource` class includes these properties.

criteria

Set the criteria property to group order or quote line items. For example, group order or quote line items based on a monthly billing frequency.

fieldValues

Set the fieldValues property to assign values to the fields to update the sales transaction.

groupAction

Set the groupAction property to group order or quote line items.

id

Set the id property to assign the ID of the sales transaction record.

method

Set the method property to specify the API request method, such as POST, PATCH, or DELETE.

type

Set the type property to assign the object type that's returned from the field describe result by using the getReferenceTo() method or from the sObject describe result by using the getSObjectType() method.

criteria

Set the criteria property to group order or quote line items. For example, group order or quote line items based on a monthly billing frequency.

Signature

```
public Map<String,ANY> criteria {get; set;}
```

Property Value

Type: Map<String,ANY>

fieldValues

Set the fieldValues property to assign values to the fields to update the sales transaction.

Signature

```
public Map<String,ANY> fieldValues {get; set;}
```

Property Value

Type: ListMap<String,ANY>

groupAction

Set the groupAction property to group order or quote line items.

You can group order or quote line items based on location, work types, or departments, if groups are enabled for your org. Groups provide a visualization of the products to view large quotes.

Signature

```
public String groupAction {get; set;}
```

Property Value

Type: String

Valid values are:

- `GroupBy`
- `Group`
- `Ungroup`
- `GroupAll`
- `DeleteGroup`

id

Set the id property to assign the ID of the sales transaction record.

Signature

```
public String id {get; set;}
```

Property Value

Type: String

method

Set the method property to specify the API request method, such as POST, PATCH, or DELETE.

Signature

```
public String method {get; set;}
```

Property Value

Type: String

type

Set the type property to assign the object type that's returned from the field describe result by using the getReferenceTo() method or from the sObject describe result by using the getSObjectType() method.

Signature

```
public Schema.SObjectType type {get; set;}
```

Property Value

Type: Schema.SObjectType

RecordWithReferenceRequest Class

Contains constructors and properties to associate a record object with a reference identifier.

Namespace

RevSalesTrxn

Example

```
RevSalesTrxn.RecordWithReferenceRequest quoteLineItemRecords = new
RevSalesTrxn.RecordWithReferenceRequest('refQuoteItem',quoteLineItemRecord);
```

RecordWithReferenceRequest Constructors

Learn more about the available constructors with the RecordWithReferenceRequest class.

RecordWithReferenceRequest Properties

Learn more about the available properties with the RecordWithReferenceRequest class.

RecordWithReferenceRequest Constructors

Learn more about the available constructors with the RecordWithReferenceRequest class.

The `RecordWithReferenceRequest` class includes these constructors.

RecordWithReferenceRequest(referenceId, record)
Creates an instance of the RecordWithReferenceRequest class to associate a record object with a reference identifier by using the referenceId and record object properties.

`RecordWithReferenceRequest(referenceId, record)`

Creates an instance of the RecordWithReferenceRequest class to associate a record object with a reference identifier by using the referenceId and record object properties.

Signature

```
public RecordWithReferenceRequest(String referenceId, RevSalesTrxn.RecordResource
record)
```

Parameters

`referenceId`
Type: String

Reference ID that maps to the subrequest response and can be used to reference the response in subsequent subrequests. You can reference the referenceId in either the body or URL of a subrequest. Use this syntax to include a reference: @{referenceId.FieldName}. See referenceId property of a composite subrequest.

`record`
Type: RevSalesTrxn.RecordResource

Record object that's defined using the `RecordResource` class.

RecordWithReferenceRequest Properties

Learn more about the available properties with the RecordWithReferenceRequest class.

The `RecordWithReferenceRequest` class has these properties.

record
Set the record property to specify the record object that's defined by using the RecordResource class.

referenceId
Set the referenceId property to specify the reference ID that maps to the subrequest response. This reference ID can be used to reference the response in subsequent subrequests.

`record`

Set the record property to specify the record object that's defined by using the RecordResource class.

Signature

```
public RevSalesTrxn.RecordResource record {get; set;}
```

Property Value

Type: RevSalesTrxn.RecordResource

`referenceId`

Set the referenceId property to specify the reference ID that maps to the subrequest response. This reference ID can be used to reference the response in subsequent subrequests.

Signature

```
public String referenceId {get; set;}
```

Property Value

Type: String

Quote and Order Capture Standard Invocable Actions

Learn more about the standard invocable actions available with Quote and Order Capture.

Create Contract Action
Create a contract from a specific quote record.

Create Order From Quote Action
Create an order from a quote record.

Create Service Document Actions
Create service documents from work orders, work order line items, or service appointments.

SEE ALSO:

Actions Developer Guide: Overview

REST API Developer Guide: Invocable Actions Standard

Create Contract Action

Create a contract from a specific quote record.

This action is available in API version 60.0 and later.

Special Access Rules

The Create Contract action is available in Developer, Enterprise, and Unlimited Editions of Revenue Cloud.

Supported REST HTTP Methods

URI

`/services/data/v63.0/actions/standard/createContract`

Formats

JSON, XML

HTTP Methods

POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`contractPriceOption`	**Type** string **Description** Optional. Determines how the contract price is set for quote line items based on the selected value. Valid values are: • `CONTRACT_HEADER_ONLY`—Creates a contract with only the header information, without using net prices or discounts. • `NET_UNIT_PRICE_ONLY`—Creates a contract specifically for quote line items with a net unit price, saving all net unit prices of the quote as contract prices. • `DISCOUNT_ONLY`—Creates contract prices specifically for quote line items with discounts, saving all discounts of the quote as contract prices. The default value is `CONTRACT_HEADER_ONLY`.
`sourceId`	**Type** string **Description** Required. ID of the quote or order that you want to create a contract from.

Outputs

Output	Details
`contractId`	**Type** string

Output	Details
	Description ID of the contract created for the specified order or quote.

Example

POST

This sample request is for the Create Contract action.

```
{
  "inputs": [
    {
      "sourceId": "0Q0RO0000003LyU",
      "contractPriceOption": "NET_UNIT_PRICE_ONLY"
    }
  ]
}
```

This sample response is for the Create Contract action.

```
[
  {
    "actionName": "createContract",
    "errors": null,
    "isSuccess": true,
    "outputValues": {
      "contractId": "800XXX123456789"
    }
  }
]
```

SEE ALSO:

Salesforce Help: Use a Custom Flow to Create Contracts

Create Order From Quote Action

Create an order from a quote record.

This action is available in API version 60.0 and later.

Special Access Rules

The Create Order From Quote action is available in Enterprise, Unlimited, and Developer Editions of Revenue Cloud.

Supported REST HTTP Methods

URI

`/services/data/v63.0/actions/standard/createOrderFromQuote`

Formats

JSON, XML

HTTP Methods

POST

Authentication

`Authorization: Bearer ***token***`

Inputs

Input	Details
`quoteRecordId`	**Type** datetime **Description** Required. ID of the quote record.

Outputs

Output	Details
`requestId`	**Type** string **Description** ID of the request.

Example

POST

This sample request is for the Create Order From Quote action.

```
{
    "inputs": [
        {
        "quoteRecordId": "0Q0D200000000DhKAI"
        }
    ]
}
```

This sample response is for the Create Order From Quote action.

```
[
    {
        "actionName": "createOrderFromQuote",
        "errors": null,
        "isSuccess": true,
```

```
        "outputValues": {
            "requestId": "143021b0-ada6-49f5-a2ea-6977a45c06a6"
        }
    }
]
```

Create Service Document Actions

Create service documents from work orders, work order line items, or service appointments.

This object is available in API version 60.0 and later.

Supported REST HTTP Methods

URI

`/services/data/v60.0/actions/standard/createServiceDocument`

Formats

JSON

HTTP Methods

POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`recordId`	**Type** string **Description** Required. The record ID of a work order, work order line item, or service appointment used to generate the service document. Create a Lightning web component to use a Custom Property Editor (CPE) to validate the `recordId` to avoid deployment issues.
`templateId`	**Type** string **Description** Required, if the `recordId` is a work order, work order line item, or service appointment. The ID of the service document template to use when generating the document.
`locale`	**Type** string **Description** Optional. Specifies the language for service document localization. The default is the user's language. Used when generating a document in a different language from the user's language. See a list of supported languages in Supported Languages.

Input	Details
	You can only input language for locale. For example, use `es` for Spanish. Using language and country, for example `es_ES` for Spanish associated with Spain, results in error.
`title`	**Type** string **Description** Optional. The value used to name the document that's generated and saved.
`documentType`	**Type** string **Description** Optional. Value that allows users to generate different types of documents by using the service. Valid values are: • `ServiceDocument`—Type of service document, such as service agreement or service contract. • `QuoteDocument`—Type of quote document, such as sales quote or service quote. • `SfsQuoteDocument`—Type of quote document for Salesforce Field Service (SFS), suitable for mobile use. This document is stored in the QuoteDocument object, and is generated through flow-based processes that link to related service documents. The default value is `ServiceDocument`.
`pdfReportId`	**Type** string **Description** Optional value corresponding to `recordId` and `templateId`. However, the value is required if you aren't generating the document from the default `pdfRecord` record. For the Document Builder feature, this is a service report ID for a report that is in progress, queued, or failed. It must be used to generate a service document from failed state.

Outputs

Inputs	Details
`pdfReportId`	**Type** string **Description** Required. The report's record ID that holds the generated PDF. For service documents, the `pdfReportId` is a service report, and the record is created if the work order, work order line item, or service appointment is passed as the `recordId`.

Usage

Sample Input

The following code sample generates a PDF of a service document with a specific `recordId` and `templateId`:

```
{
  "inputs":[
    {
      "recordId": "08pOG00000023anYAA",
      "templateId": "0M0OG0000005Na40AE",
      "locale": "en_US",
      "title": "My Awesome PDF"
    }
  ]
}
```

Quote and Order Capture Metadata API Types

Metadata API enables you to access some types and feature settings that you can customize in the user interface.

Flow for Quote and Order Capture
The flow for Quote and Order Capture represents the metadata associated with a flow. With Flow, you can create an application that takes users through a series of pages to query and update the records in the database. You can also run logic and provide branching capability based on user input to build dynamic applications.

SEE ALSO:

Metadata API Developer Guide: Understanding Metadata API

Flow for Quote and Order Capture

The flow for Quote and Order Capture represents the metadata associated with a flow. With Flow, you can create an application that takes users through a series of pages to query and update the records in the database. You can also run logic and provide branching capability based on user input to build dynamic applications.

FlowActionCall

Quote and Order Capture exposes additional actionType values for the FlowActionCall metadata type.

Field Name	Field Type	Description
`actionType`	InvocableActionType (enumeration of type string)	Required. The action type. Additional valid values for Quote and Order Capture are: • `createOrderFromQuote`—Create an order from a quote record. • `createContract`—Create a contract from a specific quote record.

Asset Lifecycle

Get insights into customer assets and increase customer lifetime value. Also, get a consolidated list of all assets that belong to an account.

EDITIONS

Available in: Lightning Experience

Available in: **Enterprise**, **Unlimited**, and **Developer** Editions of Revenue Cloud

Asset Lifecycle Standard Objects
The Asset Lifecycle data model provides objects and fields to manage details of a customer asset.

Asset Lifecycle Standard Invocable Actions
Learn more about the standard invocable actions available with Asset Lifecycle management.

Asset Lifecycle Metadata API Types
Metadata API enables you to access some types and feature settings that you can customize in the user interface.

Asset Lifecycle Business APIs
Use the Asset Lifecycle Business APIs to initiate the amendment, cancellation, or renewal of assets.

Asset Lifecycle Standard Objects

The Asset Lifecycle data model provides objects and fields to manage details of a customer asset.

Asset
Represents an item of commercial value, such as a product sold by your company or a competitor, that a customer has purchased.

AssetAction
Represents a change made to a lifecycle-managed asset. The fields can't be edited. This object is available in API version 50.0 and later.

AssetActionSource
Represents an optional way to record what transactions caused changes to lifecycle-managed assets. Use it to trace financial and other information about asset actions. This object supports Salesforce order products and work order line items, and transaction IDs from other systems. The fields can't be edited. This object is available in API version 50.0 and later.

AssetAttribute
Stores asset attributes to track and analyze asset conditions to improve their uptime. This object is available in API version 57.0 and later.

AssetContractRelationship

Represents a relationship between an asset and a contract. This object is available in API version 60.0 and later.

AssetDowntimePeriod

Represents a period during which an asset is not able to perform as expected. Downtime periods include planned activities, such as maintenance, and unplanned events, such as mechanical breakdown. This object is available in API version 49.0 and later.

AssetOwnerSharingRule

Represents the rules for sharing an Asset with users other than the owner. This object is available in API version 33.0 and later.

AssetRateCardEntry

Stores the negotiated rate card entries that are associated with an asset in Revenue Cloud. This object is available in API version 62.0 and later.

AssetRelationship

Represents a non-hierarchical relationship between assets due to an asset modification; for example, a replacement, upgrade, or other circumstance. In Subscription Management and Revenue Lifecycle Management, this object represents an asset or assets grouped in a bundle or set. This object is available in API version 41.0 and later.

AssetShare

Represents a sharing entry on an Asset. This object is available in API version 33.0 and later.

AssetStatePeriod

Represents a time span when an asset has the same quantity, amount, and monthly recurring revenue (MRR). An asset has as many asset state periods as there are changes to it (asset actions) during its lifecycle. The dashboard and related pages show the current asset state period. The fields can't be edited. This object is available in API version 50.0 and later.

AssetStatePeriodAttribute

Represents a virtual object that holds the key-value pair of the asset attribute in a specified asset state period. This object is a child object of AssetStatePeriod. This object is available in API version 60.0 and later.

AssetTag

Associates a word or short phrase with an Asset.

AssetTokenEvent

The documentation has moved to AssetTokenEvent in the *Platform Events Developer Guide*.

AssetWarranty

Defines the warranty terms applicable to an asset along with any exclusions and extensions. This object is available in API version 50.0 and later.

SEE ALSO:

Object Reference for the Salesforce Platform: Overview of Salesforce Objects and Fields

SOAP API Developer Guide: Introduction to SOAP API

Asset

Represents an item of commercial value, such as a product sold by your company or a competitor, that a customer has purchased.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), undelete(), update(), upsert()`

Fields

Field	Details
`AccountId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** (Required) ID of the Account associated with this asset. Must be a valid account ID. Required if `ContactId` isn't specified. This field is a relationship field. **Relationship Name** Account **Relationship Type** Lookup **Refers To** Account
`Address`	**Type** address **Properties** Filter, Nillable **Description** Represents the physical address or geolocation of the asset.
`AssetLevel`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The asset's position in an asset hierarchy. If the asset has no parent or child assets, its level is 1. Assets that belong to a hierarchy have a level of 1 for the root asset, 2 for the child assets of the root asset, 3 for their children, and so forth. On assets created before the introduction of this field, the asset level defaults to –1. After the asset record is updated, the asset level is calculated and automatically updated.
`AssetProvidedById`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The account that provided the asset, typically a manufacturer. This field is a relationship field.

Field	Details
	Relationship Name AssetProvidedBy **Relationship Type** Lookup **Refers To** Account
`AssetServicedById`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The account in charge of servicing the asset. This field is a relationship field. **Relationship Name** AssetServicedBy **Relationship Type** Lookup **Refers To** Account
`AssetTypeId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The asset type associated with the asset. This field is a relationship field. This field is available in API version 62.0 and later for users with the Health Cloud Appointment Management permission set. **Relationship Name** AssetType **Relationship Type** Lookup **Refers To** AssetType
`Availability`	**Type** percent **Properties** Filter, Nillable, Sort

Field	Details
	Description The percentage of expected uptime where the asset was available for use.
`AveragetimetoRepair`	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** Represents the number of hours it typically takes to repair an asset after a failure.
`AveragetimeBetweenFailure`	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** Represents the number of hours that typically elapses before the asset is likely to fail again.
`AverageUptimePerDay`	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The average number of hours per day the asset is expected to be available for use.
`City`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The city detail for the address.
`ConsequenceOfFailure`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The business impact associated with the asset's failure. Using this field, you can address the asset's health and take action using Flows. To enable this field, use Object Manager to update the field availability. Make sure that the field is visible for field-level security and for page layout. To learn more, see What Determines Field Access. The picklist values aren't predefined in orgs created before Winter '22 that aren't Field Service enabled. This field is available in API version 53.0 and later.

Field	Details
	Possible values are: • `Insignificant` • `Minor` • `Moderate` • `Major` • `Critical`
`ContactId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Required if `AccountId` isn't specified. ID of the Contact associated with this asset. Must be a valid contact ID that has an account parent (but doesn't need to match the asset's `AccountId`). This field is a relationship field. **Relationship Name** Contact **Relationship Type** Lookup **Refers To** Contact
`Country`	**Type** String **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The country detail for the address.
`CurrencyIsoCode`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Three-letter ISO 4217 currency code associated with the invoice. The default value is USD. This field is available in API version 55.0 and later. This field is available when CPQ Plus, Salesforce Billing, or Subscription Management is enabled.
`CurrentAmount`	**Type** currency

Field	Details
	Properties Filter, Nillable, Sort **Description** Reserved for future use. This field is available in API version 50.0 and later. This field is available when CPQ Plus, Salesforce Billing, or Subscription Management is enabled.
`CurrentLifecycleEndDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Represents the end of the period shown as current. System-populated field inherited from the end date of the current asset state period. If that field is empty, as with an evergreen subscription, the Current Lifecycle End Date field is also empty. This field is available in API version 50.0 and later. This field is available when CPQ Plus, Salesforce Billing, or Subscription Management is enabled.
`CurrentMrr`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The asset's monthly recurring revenue during the current asset state period. System-populated field inherited from the monthly recurring revenue on the current asset state period. If no asset state period is current, the value is `0`. Label is Current Monthly Recurring Revenue. This field is available in API version 50.0 and later. This field is available when CPQ Plus, Salesforce Billing, or Subscription Management is enabled.
`CurrentQuantity`	**Type** double **Properties** Filter, Nillable, Sort **Description** The asset's quantity during the current asset state period. System-populated field inherited from the quantity on the current asset state period. If no asset state period is current, the value is `0`. This field is available in API version 50.0 and later. This field is available when CPQ Plus, Salesforce Billing, or Subscription Management is enabled.
`Description`	**Type** textarea

Field	Details
	Properties Create, Nillable, Update **Description** Description of the asset.
`DigitalAssetStatus`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Status of digital tracking of the asset. The default picklist includes the following values: • `On` • `Off` • `Warning` • `Error`
`ExternalIdentifier`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ID of the matching record in an external system. This field is available in API version 49.0 and later.
`GeocodeAccuracy`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Accuracy level of the geocode for the address.
`HasLifecycleManagement`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** True if this asset is a lifecycle-managed asset, otherwise false. You can't switch an asset to a lifecycle-managed asset or the reverse. This field is system populated. The default value is *`false`*. This field is available in API version 50.0 and later. This field is available when CPQ Plus, Salesforce Billing, or Subscription Management is enabled.

Field	Details
`InstallDate`	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Date when the asset was installed.
`IsCompetitorProduct`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether this Asset represents a product sold by a competitor (`true`) or not (`false`). The default value is `false`. Its UI label is Competitor Asset.
`IsInternal`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates that the asset is produced or used internally (`true`) or not (`false`). The default value is `false`. Its UI label is Internal Asset.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date and time that the asset was last modified. Its UI label is Last Modified Date.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date and time that the asset was last viewed.
`Latitude`	**Type** double **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description Used with Longitude to specify the precise geolocation of the address.
`LifecycleEndDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Represents the end of the asset's lifecycle. System-populated field inherited from the end date of the final asset state period. If that field is empty, as with an evergreen subscription, the lifecycle has no end date. This field is available in API version 50.0 and later. This field is available when CPQ Plus, Salesforce Billing, or Subscription Management is enabled.
`LifecycleStartDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Represents the beginning of the asset's lifecycle. System-populated field inherited from the start date of the earliest asset state period. This field can't be edited. When a new asset action affects the start date of an asset state period, the period is deleted and a new one is generated. This field is available in API version 50.0 and later. This field is available when CPQ Plus, Salesforce Billing, or Subscription Management is enabled.
`LocationId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The asset's location. Typically, this location is the place where the asset is stored, such as a warehouse or van.
`Longitude`	**Type** double **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Used with Latitude to specify the precise geolocation of the address.
`ManufactureDate`	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The date when the asset was manufactured. This field is available from API version 49.0 and later.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** (Required) Name of the asset. Label is Asset Name.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The asset's owner. By default, the asset owner is the user who created the asset record. Its UI label is Asset Owner. This field is a relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** User
`ParentId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The asset's parent asset. Its UI label is Parent Asset. This field is a relationship field. **Relationship Name** Parent **Relationship Type** Lookup **Refers To** Asset

Field	Details
`PostalCode`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The postal code for the address.
`Price`	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** Price paid for this asset.
`PricingSource`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Pricing source to use when amending or renewing an asset. Valid values are: • `LastTransaction`—Last Transaction • `PriceBookListPrice`—Price Book or List Price Available in API version 60.0 and later.
`Product2Id`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** (Optional) ID of the Product2 associated with this asset. Must be a valid Product2 ID. Its UI label is Product. This field is a relationship field. **Relationship Name** Product2 **Relationship Type** Lookup **Refers To** Product2

Field	Details
ProductCode	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The product code of the related product.
ProductDescription	**Type** string **Properties** Filter, Sort, Nillable **Description** The product description of the related product.
ProductFamily	**Type** picklist **Properties** Filter, Group, Sort, Nillable **Description** The product family of the related product.
PurchaseDate	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Date on which this asset was purchased.
Quantity	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** Quantity purchased or installed. The Quantity field value isn't set by Customer Asset Lifecycle Management. Instead, you can populate the field as you need.
QuantityIncreasePricingType	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field	Details
	Description Specify which pricing type to use when the quantity of this asset is increased. Its UI label is Pricing Type for Quantity Increase. This field is available in API version 56.0 and later. This field is available when Subscription Management is enabled. Possible values are: • `LastNegotiatedPrice`—Available in API version 58.0 and later. • `ListPrice`
`RecordTypeId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The unique identifier for the asset. This field is a relationship field. **Relationship Name** RecordType **Relationship Type** Lookup **Refers To** RecordType
`Reliability`	**Type** percent **Properties** Filter, Nillable, Sort **Description** The percentage of expected uptime where the asset wasn't subject to unplanned downtime.
`RenewalPricingType`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The price used when renewing a subscription. Its UI label is Pricing Type for Renewal. This field is available in API version 55.0 and later. This field is available when Subscription Management is enabled. Possible values are: • `LastNegotiatedPrice` • `ListPrice`

Field	Details
`RenewalTerm`	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** With Renewal Term Unit, defines the default subscription term for renewal quotes. This field is available in API version 55.0 and later. This field is available when Subscription Management is enabled.
`RenewalTermUnit`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The unit of time for a subscription term. This field is available in API version 55.0 and later. This field is available when Subscription Management is enabled. Possible values are: • `Annual`—Available in API version 58.0 and later. —UI label is `Years`. • `Months`
`RootAssetId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** (Read only) The top-level asset in an asset hierarchy. Depending on where an asset lies in the hierarchy, its root could be the same as its parent. Its UI label is Root Asset. This field is a relationship field. **Relationship Name** RootAsset **Relationship Type** Lookup **Refers To** Asset
`SalesStoreId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** ID of the RetailStore or WebStore associated with this Asset.

Field	Details
	This field is a polymorphic relationship field. To access this field, your org must have a Salesforce Order Management license or a B2B Commerce License. This field is available in API v60.0 and later. **Relationship Name** SalesStore **Relationship Type** Lookup **Refers To** RetailStore, WebStore
`SerialNumber`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Serial number for this asset.
`State`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The state detail for the address.
`Status`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Customizable picklist of values. The default picklist includes the following values: • `Purchased` • `Shipped` • `Installed` • `Registered` • `Obsolete`
`StatusReason`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The explanation of the device status. This field is available from API version 49.0 and later. Possible values are: • `Not Ready` • `Off` • `Offline` • `Online` • `Paused` • `Standby`
`StockKeepingUnit`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The SKU assigned to the related product.
`Street`	**Type** textarea **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The street detail for the address.
`SumDowntime`	**Type** double **Properties** Filter, Nillable, Sort **Description** Accumulated downtime (planned and unplanned), determined as follows: • When only `UptimeRecordStart` is set, the sum of all downtime from `UptimeRecordStart` • When `UptimeRecordStart` and `UptimeRecordEnd` are set, the sum of all downtime from `UptimeRecordStart` to `UptimeRecordEnd` Otherwise, downtime isn't accumulated.
`SumUnplannedDowntime`	**Type** double **Properties** Filter, Nillable, Sort

Field	Details
	Description Accumulated unplanned downtime, determined as follows: • When only `UptimeRecordStart` is set, the sum of all unplanned downtime from `UptimeRecordStart` • When `UptimeRecordStart` and `UptimeRecordEnd` are set, the sum of all unplanned downtime from `UptimeRecordStart` to `UptimeRecordEnd` Otherwise, unplanned downtime isn't accumulated.
`TotalLifecycleAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total amount of revenue for the asset, including revenue from each stage in the asset lifecycle. This field is available when CPQ Plus, Salesforce Billing, or Subscription Management is enabled.
`UptimeRecordEnd`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date until which `SumDowntime` and `SumUnplannedDowntime` are accumulated.
`UptimeRecordStart`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date from which `SumDowntime` and `SumUnplannedDowntime` are accumulated.
`UsageEndDate`	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Date when usage for this asset ends or expires.
`Uuid`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The unique ID for the asset. This field is available in API version 49.0 and later.

Usage

Use this object to track products sold to customers. With asset tracking, a client application can quickly determine which products were previously sold or are currently installed at a specific account. You can also create hierarchies of up to 10,000 assets.

For example, suppose that your company wants to renew and upsell opportunities on products sold in the past. Similarly, your company can track competitive products in a customer environment where products can be replaced or swapped out.

Asset tracking is also useful for product support, providing detailed information to assist with product-specific support issues. For example, the `PurchaseDate` or `SerialNumber` can indicate whether a given product has certain maintenance requirements, including product recalls. Similarly, the `UsageEndDate` can indicate when the asset was removed from service or when a license or warranty expires.

If an application creates an Asset record, it must specify a `Name` and either an `AccountId`, `ContactId`, or both.

With REST API, use the `getRelatedListInfo` function to get information about related lists on the asset. Note that when requesting information about *`PrimaryAssets`*, the response is labeled `Related Assets`, and the response for *`RelatedAssets`* is labeled `Primary Assets`.

Associated Objects

This object has the following associated objects. If the API version isn't specified, those objects are available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

AssetChangeEvent (API version 44.0)
Change events are available for the object.

AssetFeed
Feed tracking is available for the object.

AssetHistory
History is available for tracked fields of the object.

AssetOwnerSharingRule
Sharing rules are available for the object.

AssetShare
Sharing is available for the object.

AssetAction

Represents a change made to a lifecycle-managed asset. The fields can't be edited. This object is available in API version 50.0 and later.

Supported Calls

`describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search()`

Special Access Rules

To use Customer Asset Lifecycle Management APIs, you must have the Access Customer Asset Lifecycle Management APIs permission and Read access to the Asset, Asset Action, Asset Action Source, and Asset State Period objects.

Fields

Field	Details
`ActionDate`	**Type** dateTime **Properties** Filter, Sort **Description** The date when an asset action change is recorded. This date can differ from the start date of the related asset state period. For example, suppose that a customer cancels a subscription in June, and the subscription expires in October. The date the customer cancels the subscription (June) is the action date of the asset action. The cancellation's effective date (October) is the start date of the asset state period.
`ActualTaxChange`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The rollup of actual tax from all asset action sources. This field is populated by the system. Label is **Change in Actual Tax**. This field is a calculated field.
`AdjustmentAmountChange`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The rollup of adjustment amount from all asset action sources. This field is populated by the system. Label is **Change in Adjustment Amount**. This field is a calculated field.
`Amount`	**Type** currency **Properties** Filter, Sort **Description** The delta in the total asset amount resulting from an asset action.

Field	Details
`AssetActionNumber`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The ID of the asset action. Label is **Name**.
`AssetId`	**Type** reference **Properties** Filter, Group, Sort **Description** The ID of the related lifecycle-managed asset. Label is **Asset**. This field is a relationship field. **Relationship Name** Asset **Relationship Type** Lookup **Refers To** Asset
`Category`	**Type** picklist **Properties** Filter, Group, Sort **Description** A category to apply to the asset action. In your layouts and reports, replace this optional picklist with the required Business Category picklist. Label is **Category**. Available in API version 55.0 and earlier. Possible values are: • `Cancellations` • `Cross-Sells` • `Downsells` • `Initial Sale` • `Other` • `Renewals` • `Terms And Conditions Changes` • `Transfers` • `Upsells`

Field	Details
`CategoryEnum`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** The business category of the asset action, for use in reporting. Asset action totals are broken out by the picklist values on this required field, and those totals are in turn reflected on assets. The following categories are available. They aren't customizable. Label is **Business Category**. Possible values are: • `Cancellations` • `Cross-Sells` • `Downsells` • `Initial Sale` • `Other` • `Renewals` • `Terms And Conditions Changes` • `Transfers` • `Upsells`
`EstimatedTaxChange`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The rollup of estimated tax from all asset action sources. This field is populated by the system. Label is **Change in Estimated Tax**. This field is a calculated field.
`MrrChange`	**Type** currency **Properties** Filter, Sort **Description** The delta in the asset's monthly recurring revenue resulting from an asset action. For example, suppose that the MRR during an asset state period is $200 and the next asset action adds $100. Then this field's value is $100. Label is **Change in Monthly Recurring Revenue**.
`ProductAmountChange`	**Type** currency **Properties** Filter, Nillable, Sort

Field	Details
	Description The rollup of product amount from all asset action sources. This field is populated by the system. Label is **Change in Product Amount**. This field is a calculated field.
`QuantityChange`	**Type** double **Properties** Filter, Sort **Description** The delta in the asset quantity resulting from an asset action. For example, suppose that the asset quantity during an asset state period is 20 and the next asset action adds 10. Then this field's value is 10. Label is **Change in Quantity**.
`SubtotalChange`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The rollup of subtotal from all asset action sources. This field is populated by the system. Label is **Change in Subtotal**. This field is a calculated field.
`TotalAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of the current and previous asset action amount. This field is populated by the system. This field is a calculated field.
`TotalCancellationsAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of current and previous asset action amounts categorized as `Cancellations`. This field is populated by the system.
`TotalCrossSellsAmount`	**Type** currency

Field	Details
	Properties Filter, Nillable, Sort **Description** The sum of current and previous asset action amounts categorized as `Cross-Sells`. This field is populated by the system.
`TotalDownsellsAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of current and previous asset action amounts categorized as `Downsells`. This field is populated by the system.
`TotalInitialSaleAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of current and previous asset action amounts categorized as `Initial Sale`. This field is populated by the system.
`TotalMrr`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of the monthly recurring revenue for the current and previous asset action. This field is populated by the system. Label is **Total Monthly Recurring Revenue**.
`TotalOtherAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of current and previous asset action amounts categorized as `Other`. This field is populated by the system.
`TotalQuantity`	**Type** double **Properties** Filter, Nillable, Sort

Field	Details
	Description The sum of the changes in quantity for the current and previous asset action. This field is populated by the system.
`TotalRenewalsAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of current and previous asset action amounts categorized as `Renewals`. This field is populated by the system.
`TotalTermsAndConditionsAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of current and previous asset action amounts categorized as `Terms and Conditions Changes`. This field is populated by the system. Label is **Total Terms and Conditions Changes Amount**.
`TotalTransfersAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of current and previous asset action amounts categorized as `Transfers`. This field is populated by the system.
`TotalUpsellsAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of current and previous asset action amounts categorized as `Upsells`. This field is populated by the system.
`Type`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort

Field	Details
	Description The REST API used to generate the asset action. This field is populated by the system. Possible values are: • `Cancel` • `Change` • `Generate`

AssetActionSource

Represents an optional way to record what transactions caused changes to lifecycle-managed assets. Use it to trace financial and other information about asset actions. This object supports Salesforce order products and work order line items, and transaction IDs from other systems. The fields can't be edited. This object is available in API version 50.0 and later.

Supported Calls

`describeLayout(),describeSObjects(),getDeleted(),getUpdated(),query(),retrieve(),search()`

Special Access Rules

To use Customer Asset Lifecycle Management APIs, you must have the Access Customer Asset Lifecycle Management APIs permission and Read access to the Asset, Asset Action, Asset Action Source, and Asset State Period objects.

Fields

Field	Details
`ActualTax`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The region-specific tax amount determined at time of the order.
`AdjustmentAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** An adjustment to the product amount, such as a discount.
`AssetActionId`	**Type** reference

Field	Details
	Properties Filter, Group, Sort **Description** The related asset action, that is, the change caused by an asset action source transaction. This field is a relationship field. **Relationship Name** AssetAction **Relationship Type** Lookup **Refers To** AssetAction
`AssetActionSourceNumber`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The ID of the asset action source. Label is **Name**.
`BillingReference`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The ID of the OrderItem or OrderItemDetail record that this AssetActionSource record is created for.
`Discount`	**Type** percent **Properties** Filter, Group, Nillable, Sort **Description** The discount, expressed as a percentage, that's applied to the asset. This field is available in API version 62.0 and later.
`DiscountAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The discount, expressed as currency, that's applied to the asset.

Field	Details
	This field is available in API version 62.0 and later.
EffectiveGrantDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date when the resources associated with the asset were granted. This field is available in orgs that have Revenue Cloud when Rate Management is enabled. This field is available in API version 62.0 and later.
EndDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The end date of the service or change.
EstimatedTax	**Type** currency **Properties** Filter, Nillable, Sort **Description** The estimate of the region-specific tax amount made at time of the transaction.
ExternalReference	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The ID of an asset action source transaction originating in a system outside of Salesforce.
ExternalReferenceDataSource	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** A system outside of Salesforce that contains asset action source transactions.
LegalEntityId	**Type** reference

Field	Details
	Properties Filter, Group, Nillable, Sort **Description** The ID of the legal entity record associated with the asset action source transaction. This field is a relationship field. This field is available in API version 62.0 and later. **Relationship Name** LegalEntity **Relationship Type** Lookup **Refers To** LegalEntity
`ListPrice`	**Type** currency **Properties** Filter, Nillable, Sort **Description** List price for the order product. Value is inherited from the associated PriceBookEntry upon order product creation.
`NetUnitPrice`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The final adjusted unit price, inclusive of all adjustments, but exclusive of tax. The unit price after all price adjustments are applied.
`ObligatedAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** When a line amount is prorated, this amount shows the service amount that's been consumed.
`PeriodBoundary`	**Type** picklist **Properties** Filter, Nillable, Sort

Field	Details
	Description Boundary delimiters for periods. It determines when a period starts and/or ends. Enum: anniversary, align to calendar, end of period, day of period.
`PeriodBoundaryDay`	**Type** int **Properties** Filter, Nillable, Sort **Description** The number specifying the day number when Period Boundary is a specific day in a week/month/year. It only applies when PeriodBoundary is set to "day of period."
`PeriodBoundaryStartMonth`	**Type** picklist **Properties** Filter, Nillable, Sort **Description** Field is populated based on input in the StartDate, PeriodBoundary, and PeriodBoundaryDay when BillingFrequency2 is Annual or by manual user entry. Possible values are: 1-January 2-February 3-March 4-April 5-May 6-June 7-July 8-August 9-September 10-October 11-November 12-December
`PricebookEntryId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** PricebookEntry is used as a lookup for price information in order to pre-populate OrderItem's ListPrice and UnitPrice.

Field	Details
PricingTermCount	**Type** double **Properties** Filter, Nillable, Sort **Description** Number of pricing terms is this subscription product.
ProductAmount	**Type** currency **Properties** Filter, Nillable, Sort **Description** The product amount after the asset action source transaction.
ProductSellingModelId	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** Specifies the product selling model type. Foreignkey to ProductSellingModel entity.
ProrationPolicyId	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The Id of the ProrationPolicy.
Quantity	**Type** double **Properties** Filter, Nillable, Sort **Description** The product quantity or the change in product quantity after the asset action source transaction.
ReferenceEntityItemId	**Type** reference **Properties** Filter, Group, Nillable, Sort

Field	Details
	Description The ID of an asset action source transaction originating in Salesforce. The transaction can be an order product or a work order line item. This field is a polymorphic relationship field. **Relationship Name** ReferenceEntityItem **Relationship Type** Lookup **Refers To** OrderItem, WorkOrderLineItem
`SegmentIdentifier`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The ID of the ramp segment associated with the asset action source transaction. This field is available in API version 62.0 and later.
`StartDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The start date of the service or change.
`Subtotal`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of the product amount and the adjustment amount. This field is a calculated field.
`TaxTreatmentId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** Lookup to Tax Treatment entity. It's used to calculate tax.

Field	Details
`TotalPrice`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Calculated by the pricing engine for ARC. Summation of TotalAdjustmentAmount plus TotalLineAmount for this item.
`TotalLineAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The price of the line before any price adjustments were applied. SalesTransactionItem: ProratedStartingTotal / StartingPriceTotal. Note: TotalPrice is computed using the UnitPrice, which includes discounts (price adjustments), while TotalLineAmount doesn't include price adjustments.
`TransactionDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date of a source transaction, such as an order date.
`UnitPrice`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The unit price of the item before any discounts or tax calculation.

AssetAttribute

Stores asset attributes to track and analyze asset conditions to improve their uptime. This object is available in API version 57.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `query()`, `update()`, `upsert()`

Special Access Rules

Field Service must be enabled.

Fields

Field	Details
`AssetId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The ID of the asset. This field is a relationship field. **Relationship Name** Asset **Relationship Type** Lookup **Refers To** Asset
`AttributeDefinitionId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The ID of the attribute definition for this asset attribute. This field is a relationship field. **Relationship Name** AttributeDefinition **Relationship Type** Lookup **Refers To** AttributeDefinition
`AttributeName`	**Type** string **Properties** Filter, Group, Sort **Description** The name given to the asset attribute in the UI by the user.
`AttributePicklistValueId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The ID of the attribute picklist value if the attribute is a picklist type. This field is a relationship field. **Relationship Name** AttributePicklistValue **Relationship Type** Lookup **Refers To** AttributePicklistValue
`AttributeValue`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Stores the value of an asset attribute, for example 5-TB storage .
`ExternalId`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** An auto-generated ID of the attribute record saved in an external system (for example an HBase database).

Usage

Add asset descriptors to the AssetAttribute object instead of creating multiple custom attributes on an asset. This helps scale to a high asset volume in the system.

AssetContractRelationship

Represents a relationship between an asset and a contract. This object is available in API version 60.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

This object is available in Enterprise, Unlimited, and Developer Editions of Revenue Cloud.

Fields

Field	Details
AssetId	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The ID of the asset related to the contract. This field is a relationship field. **Relationship Name** Asset **Relationship Type** Lookup **Refers To** Asset
ContractId	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The ID of the contract related to the asset. This field is a relationship field. **Relationship Name** Contract **Relationship Type** Lookup **Refers To** Contract
EndDate	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** The end date and time of the relationship between contract and asset.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort

Field	Details
	Description The timestamp when the current user last accessed this record, a record related to this record, or a list view. The associated UI label is **Last Modified Date**.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed this record or list view. If this value is null, the user accessed this record or list view (`LastReferencedDate`) but didn't view it.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The auto-generated number assigned to AssetContractRelationship. (Read Only)
`StartDate`	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** The start date and time of the relationship between contract and asset.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

AssetContractRelationshipFeed
Feed tracking is available for the object.

AssetContractRelationshipHistory
History is available for tracked fields of the object.

AssetDowntimePeriod

Represents a period during which an asset is not able to perform as expected. Downtime periods include planned activities, such as maintenance, and unplanned events, such as mechanical breakdown. This object is available in API version 49.0 and later.

Supported Calls

`create(),delete(),describeLayout(),describeSObjects(),getDeleted(),getUpdated(),query(),retrieve(),search(),undelete(),update(),upsert()`

Fields

Field	Details
`AssetDowntimePeriodNumber`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The unique number of this asset downtime period record.
`AssetId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The ID of the asset this asset downtime period record is for.
`Description`	**Type** textarea **Properties** Create, Nillable, Update **Description** The description of this asset downtime period.
`DowntimeType`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The type of this asset downtime period. Possible values are: • `Planned` • `Unplanned`
`EndTime`	**Type** dateTime **Properties** Create, Filter, Sort, Update

Field	Details
	Description The time this asset downtime period ended.
`IsExcluded`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Whether this asset downtime period is excluded from the calculation of accumulated downtime and accumulated unplanned downtime, and therefore not included in availability and reliability calculations.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed a record related to this record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed this record. If this value is null, this record might only have been referenced (`LastReferencedDate`) and not viewed.
`StartTime`	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** The time this asset downtime period started.

AssetOwnerSharingRule

Represents the rules for sharing an Asset with users other than the owner. This object is available in API version 33.0 and later.

Note: To enable access to this object for your org, contact Salesforce customer support. However, we recommend that you instead use Metadata API to programmatically update owner sharing rules because it triggers automatic sharing rule recalculation. The SharingRules Metadata API type is enabled for all orgs.

Supported Calls

`create(),delete(),describeSObjects(),getDeleted(),getUpdated(),query(),retrieve(),update(),upsert()`

Special Access Rules

Customer Portal users can't access this object.

Fields

Field	Details
`AssetAccessLevel`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** A value that represents the type of sharing being allowed. The possible values are: • `Read` • `Edit`
`Description`	**Type** textarea **Properties** Create, Filter, Nillable, Sort, Update **Description** A description of the sharing rule. Maximum size is 1000 characters.
`DeveloperName`	**Type** string **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Corresponds to **Rule Name** in the user interface. Note: When creating large sets of data, always specify a unique `DeveloperName` for each record. If no `DeveloperName` is specified, performance may slow while Salesforce generates one for each record.

Field	Details
`GroupId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The ID representing the source group. Cases owned by users in the source group trigger the rule to give access.
`Name`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** Label of the sharing rule as it appears in the user interface. Limited to 80 characters. Corresponds to **Label** on the user interface.
`UserOrGroupId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The ID representing the target user or group. Target users or groups are given access.

Usage

Use this object to manage the sharing rules for assets. General sharing uses this object.

AssetRateCardEntry

Stores the negotiated rate card entries that are associated with an asset in Revenue Cloud. This object is available in API version 62.0 and later.

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.

Supported Calls

`describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

This object is available in orgs where Revenue Cloud is enabled.

Fields

Field	Details
AssetId	**Type** reference **Properties** Filter, Group, Sort **Description** The ID of the asset rate card entry record. This field is a relationship field. **Relationship Name** Asset **Relationship Type** Master-detail **Refers To** Asset (the master object)
EndDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date when the rate card's time period becomes inactive. The rate card becomes inactive at 11:59:00 PM on the end date.
Name	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** An auto-generated number assigned to the asset rate card entry. Read-only.
NegotiatedRate	**Type** double **Properties** Filter, Sort **Description** The base negotiated rate used to charge overage consumption.
RateCardEntryId	**Type** reference **Properties** Filter, Group, Nillable, Sort

Field	Details
	Description The ID of the rate card entry record containing the catalog rates that's associated with the asset rate card entry. This field is a relationship field. **Relationship Name** RateCardEntry **Refers To** RateCardEntry
`RateCardId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of the rate card record that's associated with the asset rate card entry. This field is a relationship field. **Relationship Name** RateCard **Refers To** RateCard
`RateUnitOfMeasureId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of the unit of measure record that's associated with the asset rate card entry. This field is a relationship field. **Relationship Name** RateUnitOfMeasure **Refers To** UnitOfMeasure
`StartDate`	**Type** dateTime **Properties** Filter, Sort **Description** The date when the rate card's time period becomes active. The rate card becomes active at 12:00:00 AM on the start date.

Field	Details
`UsageResourceId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of the usage resource record that's associated with the asset rate card entry. This field is a relationship field. **Relationship Name** UsageResource **Refers To** UsageResource

AssetRelationship

Represents a non-hierarchical relationship between assets due to an asset modification; for example, a replacement, upgrade, or other circumstance. In Subscription Management and Revenue Lifecycle Management, this object represents an asset or assets grouped in a bundle or set. This object is available in API version 41.0 and later.

Asset relationships appear in the Primary Assets and Related Assets related lists on asset records in the UI.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

Some fields are available only in Subscription Management and Revenue Cloud. Field availability is noted in the field detail column.

Fields

Field Name	Details
`AssetId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The unique identifier of the new asset, which is the asset that is taking the place of the existing asset. This field is a relationship field. **Relationship Name** Asset

Field Name	Details
	Relationship Type Lookup **Refers To** Asset
`AssetRelationshipNumber`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** An auto-generated number identifying the asset relationship.
`AssetRole`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** Describes the position of the main asset relative to the other assets in the relationship. This field is available in API version 58.0 and later. This field is available in orgs with Subscription Management or Revenue Cloud. Possible values are: • `Add-on`—The main asset is an add-on. • `Bundle`—The main asset is the bundle parent. • `Set`—The asset is the main asset in the set.
`CurrencyIsoCode`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Three-letter ISO 4217 currency code associated with the asset. The default value is USD.
`FromDate`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date when the new asset was installed.

Field Name	Details
GroupingKey	**Type** string **Properties** Filter, Group, idLookup, Nillable, Sort **Description** Read-only field used to indicate the bundle that an asset belongs to. For example, if two assets have the same GroupingKey value, then it means that the assets are bundled together. This field is available in API v.60.0 and later. This field is available in orgs with Subscription Management or Revenue Cloud.
ProductRelationshipTypeId	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The unique identifier of the record that describes the relationship between the main and associated assets. This field is available in API version 58.0 and later. This field is available in orgs with Subscription Management. This field is a relationship field. **Relationship Name** ProductRelationshipType **Relationship Type** Lookup **Refers To** ProductRelationshipType
ProductRelatedComponent	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The product related component that's associated with the asset relationship. This field is a relationship field. This field is available in API 60.0 and later in Revenue Cloud. **Relationship Name** ProductRelatedComponent **Relationship Type** Lookup

Field Name	Details
	Refers To ProductRelatedComponent
`RelatedAssetId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The existing asset that is being modified. This field is a relationship field. **Relationship Name** RelatedAsset **Relationship Type** Lookup **Refers To** Asset
`RelatedAssetPricing`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** Specifies whether the price of the related asset is included in the bundle price. Valid values are: • `IncludedInBundlePrice` • `NotIncludedInBundlePrice` This field is available in API version 59.0 and later in Revenue Cloud.
`RelatedAssetQtyScaleMethod`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** Specifies how the quantity of the related asset changes relative to the quantity of the parent asset. Valid values are: • `Constant` • `Proportional` This field is available in API version 59.0 and later in Revenue Cloud.
`RelatedAssetRole`	**Type** picklist

Field Name	Details
	Properties Filter, Group, Nillable, Restricted picklist, Sort **Description** Describes the position of the associated asset relative to other assets in the relationship. This field is available in API version 58.0 and later. This field is available in orgs with Subscription Management or Revenue Cloud. Valid values are: • `Add-on`—The main asset is an add-on. • `Bundle`—The main asset is the bundle parent. • `Set`—The asset is the main asset in the set. • `Simple`—The asset is purchased individually and isn't associated with variations. • `Variation Parent`——The main asset is the variation parent.
`RelationshipType`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Sort, Update **Description** The type of relationship between the existing asset and the new asset. This field comes with three values—Replacement, Upgrade, and Crossgrade—, but you can create more values in Setup. Possible values are: • `Crossgrade`—The new asset is a crossgrade of an existing asset. For example, changing a subscription to a plan with the same service, but that runs for a longer amount of time. • `Replacement`—The new asset is replacing an existing asset. For example, a customer's faulty widget that was under warranty is being replaced with a new one. • `Upgrade`—The new asset is an upgrade of an existing asset. For example, upgrading a customer's existing subscription plan to a new plan with more services. The default value is `Replacement`.
`ToDate`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date when the modified asset is uninstalled.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

AssetRelationshipChangeEvent (API version 62.0)
Change events are available for the object.

AssetRelationshipFeed
Feed tracking is available for the object.

AssetRelationshipHistory
History is available for tracked fields of the object.

AssetRelationshipOwnerSharingRule (API version 58.0)
Sharing rules are available for the object.

AssetRelationshipShare (API version 58.0)
Sharing is available for the object.

AssetShare

Represents a sharing entry on an Asset. This object is available in API version 33.0 and later.

You can only create, edit, and delete sharing entries for standard objects whose `RowCause` field is set to `Manual`. Sharing entries for standard objects with different `RowCause` values are created as a result of your Salesforce org's sharing configuration and are read-only. For some sharing mechanisms, such as sharing sets, sharing entries aren't stored at all.

Note: While Salesforce currently maintains read-only sharing entries for multiple sharing mechanisms, it's possible that we'll stop storing certain share records to improve performance. As a best practice, don't create customizations that rely on the availability of these sharing entries. Any changes to sharing behavior will be communicated before they occur.

Supported Calls

`describeSObjects(), query(), retrieve()`

Special Access Rules

Customer Portal users can't access this object.

Fields

The properties available for some fields depend on the default organization-wide sharing settings. The properties listed are true for the default settings of such fields.

Field	Details
`AssetAccessLevel`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Level of access that the User or Group has to the Asset. The possible values are:

Field	Details
	• `Read` • `Edit` • `All` This value is not valid for creating or deleting records. This field must be set to an access level that is higher than the organization's default access level for cases.
`AssetId`	**Type** reference **Properties** Filter, Group, Sort **Description** ID of the Asset associated with this sharing entry. This field can't be updated. This is a relationship field. **Relationship Name** Asset **Relationship Type** Lookup **Refers To** Asset
`IsDeleted`	**Type** boolean **Properties** Defaulted on create, Filter **Description** Indicates whether the object has been moved to the Recycle Bin (`true`) or not (`false`). Label is **Deleted**.
`RowCause`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** Reason that this sharing entry exists. If you're creating a sharing entry, the only permitted value is `Manual`. If no value is specified, the field defaults to `Manual`. All other `RowCause` values are read-only. After the sharing entry is created, this field can't be edited. Valid values include: • `Manual`—The User or Group has access because a user with "All" access manually shared the Asset with them. • `Owner`—The User is the owner of the Asset. • `Rule`—The User or Group has access via an Asset sharing rule.

Field	Details
	• `GuestRule`—The User or Group has access via an Asset guest user sharing rule. • `LpuImplicit`—The User has access to records owned by high-volume Experience Cloud site users via a share group.
`UserOrGroupId`	**Type** reference **Properties** Filter, Group, Sort **Description** ID of the User or Group that has been given access to the Asset. This field can't be updated. This is a polymorphic relationship field. **Relationship Name** UserOrGroup **Relationship Type** Lookup **Refers To** Group, User

Usage

This object allows you to determine which users and groups can view and edit Asset records owned by other users.

If you attempt to create a new record that matches an existing record, request updates any modified fields and returns the existing record.

AssetStatePeriod

Represents a time span when an asset has the same quantity, amount, and monthly recurring revenue (MRR). An asset has as many asset state periods as there are changes to it (asset actions) during its lifecycle. The dashboard and related pages show the current asset state period. The fields can't be edited. This object is available in API version 50.0 and later.

Supported Calls

`describeLayout(),describeSObjects(),getDeleted(),getUpdated(),query(),retrieve(),search()`

Special Access Rules

To use Customer Asset Lifecycle Management APIs, you must have the Access Customer Asset Lifecycle Management APIs permission and Read access to the Asset, Asset Action, Asset Action Source, and Asset State Period objects.

Fields

Field	Details
`Amount`	**Type** currency **Properties** Filter, Sort **Description** An asset's total amount during an asset state period.
`AssetId`	**Type** reference **Properties** Filter, Group, Sort **Description** The asset related to an asset state period. Label is **Asset**. This field is a relationship field. **Relationship Name** Asset **Relationship Type** Lookup **Refers To** Asset
`AssetStatePeriodNumber`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The ID of the asset state period. Label is **Name**.
`BindingInstanceTargetId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of a custom product target for a usage-based quote line item, order Item, or asset allocation. This field is a polymorphic relationship field. **Relationship Name** BindingInstanceTarget

Field	Details
	Refers To Account, Asset, BindingObjectCustomExt, Contract
`EndDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The end date and time of an asset state period. On an asset that is an evergreen subscription, the last asset state period has no end date.
`Mrr`	**Type** currency **Properties** Filter, Sort **Description** An asset's monthly recurring revenue during an asset state period.
`Quantity`	**Type** double **Properties** Filter, Sort **Description** The total quantity of an asset during an asset state period.
`RampIdentifier`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The ramp record used to group order item segments for this asset state period. This field is available in orgs that have Revenue Cloud when the Ramp Deals setting is enabled. This field is available in API version 62.0 and later.
`SegmentIdentifier`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The order item segment for this asset state period. This field is available in orgs that have Revenue Cloud when the Ramp Deals setting is enabled.

Field	Details
	This field is available in API version 62.0 and later.
`SegmentName`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The name of the order item segment for this asset state period. This field is available in orgs that have Revenue Cloud when the Ramp Deals setting is enabled. This field is available in API version 62.0 and later.
`SegmentType`	**Type** picklist **Properties** Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort **Description** The period for the order item segment for this asset state period. Valid values are: • `Custom` • `Free Trial` • `Yearly` The default value is `Yearly`. This field is available in orgs that have Revenue Cloud when the Ramp Deals setting is enabled. This field is available in API version 62.0 and later.
`StartDate`	**Type** dateTime **Properties** Filter, Sort **Description** The start date and time of an asset state period.

AssetStatePeriodAttribute

Represents a virtual object that holds the key-value pair of the asset attribute in a specified asset state period. This object is a child object of AssetStatePeriod. This object is available in API version 60.0 and later.

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.

Supported Calls

`describeLayout(), describeSObjects(), query(), retrieve()`

Special Access Rules

This object is available in Enterprise, Unlimited, and Developer Editions of Revenue Cloud.

Fields

Field	Details
`AssetStatePeriodId`	**Type** reference **Properties** Filter, Group, Sort **Description** The asset state period that's associated with the asset attribute. This field is a relationship field. **Relationship Name** AssetStatePeriod **Relationship Type** Master-detail **Refers To** AssetStatePeriod (the master object)
`AttributeDefinitionId`	**Type** reference **Properties** Filter, Group, Sort **Description** The attribute definition that's associated with the asset state period attribute. This field is a relationship field. **Relationship Name** AttributeDefinition **Relationship Type** Lookup **Refers To** AttributeDefinition
`AttributeName`	**Type** string **Properties** Filter, Group, idLookup, Nillable, Sort

Field	Details
	Description The name of the asset attribute.
`AttributePicklistValueId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The value specified in the picklist type field that corresponds to the attribute in the AttributePicklistValue object. This field is a relationship field. **Relationship Name** AttributePicklistValue **Relationship Type** Lookup **Refers To** AttributePicklistValue
`AttributeValue`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The value of the asset state period attribute. For example, a shirt can have the value of `blue`, which indicates the shirt's color, or it can have the value of `small`, which indicates the shirt's size.

AssetTag

Associates a word or short phrase with an Asset.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`

Fields

Field Name	Details
`ItemId`	**Type** reference

Field Name	Details
	Properties Create, Filter **Description** ID of the tagged item.
`Name`	**Type** string **Properties** Create, Filter **Description** Name of the tag. If this value does not already exist, a new TagDefinition is created and becomes the parent of this Tag object. Otherwise, a TagDefinition with the same name becomes the parent of this Tag object. Parent relationships are created automatically.
`TagDefinitionId`	**Type** reference **Properties** Filter **Description** ID of the parent TagDefinition object that owns the tag.
`Type`	**Type** picklist **Properties** Create, Filter, Restricted picklist **Description** Defines the visibility of a tag. Valid values: • `Public`—The tag can be viewed and manipulated by all users in an organization. • `Personal`—The tag can be viewed or manipulated only by a user with a matching `OwnerId`.

Usage

AssetTag stores the relationship between its parent TagDefinition and the Asset being tagged. Tag objects act as metadata, allowing users to describe and organize their data.

When a tag is deleted, its parent TagDefinition will also be deleted if the name is not being used; otherwise, the parent remains. Deleting a TagDefinition sends it to the Recycle Bin, along with any associated tag entries.

AssetTokenEvent

The documentation has moved to AssetTokenEvent in the *Platform Events Developer Guide*.

AssetWarranty

Defines the warranty terms applicable to an asset along with any exclusions and extensions. This object is available in API version 50.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`AssetId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The ID of the asset this warranty term applies to.
`AssetWarrantyNumber`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The identifier of the asset warranty record.
`EndDate`	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The date on which this warranty term expires.
`ExchangeType`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The type of exchange offered by this warranty term.
`Exclusions`	**Type** textarea

Field	Details
	Properties Create, Nillable, Update **Description** Description of any exclusions.
`ExpensesCovered`	**Type** percent **Properties** Create, Filter, Nillable, Sort, Update **Description** The percentage of expenses covered.
`ExpensesCoveredEndDate`	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The date on which cover for expenses ends.
`IsTransferable`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Defines whether the warranty term can be transferred to a new owner.
`LaborCovered`	**Type** percent **Properties** Create, Filter, Nillable, Sort, Update **Description** The percentage of labor covered.
`LaborCoveredEndDate`	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The date on which cover for labor ends.
`LastReferencedDate`	**Type** dateTime

Field	Details
	Properties Filter, Nillable, Sort **Description** The date when the asset warranty term was last modified. Its label in the user interface is `Last Modified Date`.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date when the asset warranty term was last viewed.
`PartsCovered`	**Type** percent **Properties** Create, Filter, Nillable, Sort, Update **Description** The percentage of parts covered.
`PartsCoveredEndDate`	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The date on which cover for parts ends.
`Pricebook2Id`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ID of the price book item associated with this asset warranty term.
`StartDate`	**Type** date **Properties** Create, Filter, Group, Sort, Update **Description** The date on which cover under this warranty term starts.

Field	Details
WarrantyTermId	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ID of the warranty term this asset warranty term extends.
WarrantyType	**Type** picklist **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The type of the warranty.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

AssetWarrantyChangeEvent
Change events are available for the object.

Asset Lifecycle Standard Invocable Actions

Learn more about the standard invocable actions available with Asset Lifecycle management.

Create or Update Asset From Order Action
Create an asset for each order item in the specified order. New assets are created for a new order. Modify existing assets for change order requests, such as a renewal or a cancellation.

Create or Update Asset From Order Item Action
Create assets from individual order items within an order. Track assets after the individual line items of an order reach a certain stage in their lifecycle, such as submitted, fulfilled, or provisioned. If the order item is part of a renewal, an amendment, or a cancellation, existing assets are changed.

Initiate Amendment Action
Initiate the amendment of an asset.

Initiate Cancellation Action
Initiate the cancellation of an asset.

Initiate Renewal Action

Initiate the renewal of an asset.

SEE ALSO:

Actions Developer Guide: Overview

REST API Developer Guide: Invocable Actions Standard

Create or Update Asset From Order Action

Create an asset for each order item in the specified order. New assets are created for a new order. Modify existing assets for change order requests, such as a renewal or a cancellation.

This action is available in API version 60.0 and later.

Special Access Rules

You need the Assetize Order permission set to use this invocable action.

Supported REST HTTP Methods

URI

`/services/data/v63.0/actions/standard/createOrUpdateAssetFromOrder`

Formats

JSON, XML

HTTP Methods

POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`orderId`	**Type** string **Description** Required. ID of the order.

Outputs

Output	Details
`requestId`	**Type** string

Output	Details
	Description ID of the request to create an asset.

Example

POST

This sample request is for the Create or Update Asset From Order action.

```
{
  "inputs": [
    {
      "orderId": "801DE000000oJfAYAU"
    }
  ]
}
```

This sample response is for the Create or Update Asset From Order action.

```
[
    {
        "actionName": "createOrUpdateAssetFromOrder",
        "errors": null,
        "isSuccess": true,
        "outputValues": {
            "requestId": "3b89392d-6987-40d9-9190-d18fdb5cb849"
        }
    }
]
```

Create or Update Asset From Order Item Action

Create assets from individual order items within an order. Track assets after the individual line items of an order reach a certain stage in their lifecycle, such as submitted, fulfilled, or provisioned. If the order item is part of a renewal, an amendment, or a cancellation, existing assets are changed.

This action is available in API version 60.0 and later.

Special Access Rules

You need the Assetize Order permission set to use this invocable action.

Supported REST HTTP Methods

URI

`/services/data/v63.0/actions/standard/createOrUpdateAssetFromOrderItem`

Formats

JSON, XML

HTTP Methods

POST

Authentication

`Authorization: Bearer ***token***`

Inputs

Input	Details
`orderId`	**Type** string **Description** Required. ID of the order.

Outputs

Output	Details
`requestId`	**Type** string **Description** ID of the request to create an asset.

Example

POST

This sample request is for the Create or Update Asset From Order Item action.

```
{
    "inputs": [
        {
            "orderItemIds": ["802SG000002HixxYAC"]
        }
    ]
}
```

This sample response is for the Create or Update Asset From Order Item action.

```
{
  "actionName": "createOrUpdateAssetFromOrderItem",
  "errors": null,
  "isSuccess": true,
  "outputValues": {
    "requestId": "b2a2b4b9-845b-4078-980a-759308389604"
  },
```

```
  "version": 1
}
```

Initiate Amendment Action

Initiate the amendment of an asset.

This action is available in API version 60.0 and later.

Supported REST HTTP Methods

URI

`/services/data/v63.0/actions/standard/initiateAmendment`

Formats

JSON, XML

HTTP Methods

POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`amendAssetIds`	**Type** string **Description** Required. The IDs of the assets that you want to amend.
`amendContractId`	**Type** string **Description** The ID of the contract record to be synced with the amendment.
`amendStartDate`	**Type** datetime **Description** Required. Effective start date of the amendment.
`amendOpportunityId`	**Type** string

Input	Details
	Description The ID of the Opportunity record to be synced with the amendment quote.
`amendOutputType`	**Type** string **Description** Required. Type of amendment record to create.
`quantityChange`	**Type** double **Description** Required. Quantity to add to or reduce from the asset's existing quantity.

Outputs

Output	Details
`amendRecordId`	**Type** string **Description** The ID of the amendment record that's created.
`requestIdentifier`	**Type** string **Description** Request ID that's used to track the async request.

Example

POST

This sample request is for the Initiate Amendment action.

```
{
    "inputs": [
        {
            "amendAssetIds": ["02iI8000000HPzXIAW"],
            "amendStartDate": "2023-10-21T00:00:00.000Z",
            "quantityChange": 5,
            "amendOutputType": "Quote",
```

```
            "amendContractId" : "contract ID",
            "amendOpportunityId" : "Opportunity ID"
        }
    ]
}
```

This sample response is for the Initiate Amendment action.

```
[
    {
        "actionName": "initiateAmendment",
        "errors": null,
        "isSuccess": true,
        "outputValues": {
            "record_id": "0Q0xx0000004NsSCAU",
            "requestIdentifier": "16Pxx0000004NIy"
        },
        "version": 1
    }
]
```

Initiate Cancellation Action

Initiate the cancellation of an asset.

This action is available in API version 60.0 and later.

Supported REST HTTP Methods

URI

`/services/data/v63.0/actions/standard/initiateCancellation`

Formats

JSON, XML

HTTP Methods

POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`cancelAssetIds`	**Type** string **Description** Required. The IDs of the assets that you want to cancel.

Input	Details
`cancelContractId`	**Type** string **Description** The ID of the contract record to be synced with the cancellation.
`cancelOpportunityId`	**Type** string **Description** The ID of the Opportunity record to be synced with the cancellation quote.
`cancelOutputType`	**Type** string **Description** Required. Type of cancellation record to create.
`cancelStartDate`	**Type** datetime **Description** Required. Effective date of the cancellation.

Outputs

Output	Details
`cancelRecordId`	**Type** string **Description** The ID of the cancellation record that's created.
`requestIdentifier`	**Type** string **Description** Request ID that's used to track the async request.

Example

POST

This sample request is for the Initiate Cancellation action.

```
{
    "inputs": [
        {
            "cancelAssetIds": [
                "02iI8000000Lc5fIAC"
            ],
            "cancelStartDate": "2023-11-09T00:00:00",
            "cancelOutputType": "Quote",
            "cancelContractId" : "contract ID",
            "cancelOpportunityId" : "Opportunity ID"
        }
    ]
}
```

This sample response is for the Initiate Cancellation action.

```
[
    {
        "actionName": "initiateCancellation",
        "errors": null,
        "isSuccess": true,
        "outputValues": {
            "record_id": "0Q0xx0000004P32CAE",
            "requestIdentifier": "16Pxx0000004OTY"
        },
        "version": 1
    }
]
```

Initiate Renewal Action

Initiate the renewal of an asset.

This action is available in API version 60.0 and later.

Supported REST HTTP Methods

URI

`/services/data/v63.0/actions/standard/initiateRenewal`

Formats

JSON, XML

HTTP Methods

POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`renewAssetIds`	**Type** string **Description** Required. The IDs of the assets that you want to renew.
`renewContractId`	**Type** string **Description** The ID of the contract record to be synced with the renewal.
`renewEndDate`	**Type** datetime **Description** Effective end date of the renewal. Available in API version 62.0 and later.
`renewOpportunityId`	**Type** string **Description** The ID of the Opportunity record to be synced with the renewal quote.
`renewOutputType`	**Type** string **Description** Required. Type of renewal record to create.
`renewStartDate`	**Type** datetime **Description** Effective start date of the renewal. Available in API version 62.0 and later.

Outputs

Output	Details
`renewRecordId`	**Type** string **Description** The ID of the amendment record that's created.
`requestIdentifier`	**Type** string **Description** Request ID that's used to track the async request.

Example

POST

This sample request is for the Initiate Renewal action.

```
{
    "inputs": [
        {
            "renewAssetIds": ["02ixx0000004LMwAAM"],
            "renewOutputType": "Quote",
            "renewContractId" : "contract ID",
            "renewOpportunityId" : "Opportunity ID",
            "renewStartDate": "2023-10-21T00:00:00.000Z",
            "renewEndDate": "2024-10-21T00:00:00.000Z"
        }
    ]
}
```

This sample response is for the Initiate Renewal action.

```
[
    {
        "actionName": "initiateRenewal",
        "errors": null,
        "isSuccess": true,
        "outputValues": {
            "record_id": "0Q0xx0000004P32CAE",
            "requestIdentifier": "16Pxx0000004OTY"
        },
        "version": 1
    }
]
```

Asset Lifecycle Metadata API Types

Metadata API enables you to access some types and feature settings that you can customize in the user interface.

Flow for Asset Lifecycle

Represents the metadata associated with a flow. With Flow, you can create an application that navigates users through a series of screens to query and update records in the database. You can also execute logic and provide branching capability based on user input to build dynamic applications.

SEE ALSO:

Metadata API Developer Guide: Understanding Metadata API

Flow for Asset Lifecycle

Represents the metadata associated with a flow. With Flow, you can create an application that navigates users through a series of screens to query and update records in the database. You can also execute logic and provide branching capability based on user input to build dynamic applications.

FlowActionCall

Asset Lifecycle exposes additional actionType values for the FlowActionCall Metadata type.

Field Name	Field Type	Description
`actionType`	InvocableActionType (enumeration of type string)	Required. The action type. Additional valid values only for Asset Lifecycle include: • `createOrUpdateAssetFromOrder`—Create an asset for each order item in the specified order. New assets are created for a new order. Modify existing assets for change order requests, such as a renewal or a cancellation. • `createOrUpdateAssetFromOrderItem`—Create assets from individual order items within an order. Track assets after the individual line items of an order reach a certain stage in their lifecycle, such as submitted, fulfilled, or provisioned. If the order item is part of a renewal, an amendment, or a cancellation, existing assets are changed. • `initiateAmendment`—Initiate the amendment of an asset. • `initiateRenewal`—Initiate the renewal of an asset. • `initiateCancellation`—Initiate the cancellation of an asset.

Asset Lifecycle Business APIs

Use the Asset Lifecycle Business APIs to initiate the amendment, cancellation, or renewal of assets.

Resources

Learn more about the available Asset Lifecycle resources.

Request Bodies

Learn more about the available request bodies of Asset Lifecycle Business APIs.

Response Bodies

Learn more about the available response bodies of Asset Lifecycle Business APIs.

SEE ALSO:

https://help.salesforce.com/s/articleView?id=sf.qocal_manage_assets_in_revenue_lifecycle_management.htm

Resources

Learn more about the available Asset Lifecycle resources.

Asset Amendment (POST)

Initiate the amendment of a quote or an order.

Asset Cancellation (POST)

Initiate the cancellation of an asset.

Asset Renewal (POST)

Initiate the renewal of an asset.

Asset Amendment (POST)

Initiate the amendment of a quote or an order.

Special Access Rules

To use this API, you need the InitiateAmend API permission set.

Resource

```
/connect/revenue-management/assets/actions/amend
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/revenue-management/assets/actions/amend
```

Available version

62.0

HTTP methods

POST

Request body for POST

JSON example

```
{
  "assetIds": [
    "02iSG0000003NMhYAM",
    "02iSG0000006DvSYAU"
  ],
  "amendmentStartDate": "2023-10-04T00:00:00",
  "contractId": "800SG00000CFpepYAD",
  "opportunityId": "006SG000004W5tVYAS",
  "outputRecordId": "801SG00000DX1jWYAT",
  "outputRecordType": "Quote",
```

```
  "quantityChange": 5
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
assetIds	String[]	IDs of the assets that you want to add to the amendment record.	Required	62.0
amendmentStartDate	String	Start date of the amendment.	Required	62.0
contractId	String	ID of the Contract record that you want to sync with the amendment quote.	Optional	62.0
opportunityId	String	ID of the Opportunity record that you want to sync with the amendment quote.	Optional	62.0
outputRecordId	String	ID of the quote or order record that you want to add the assets to.	Optional	62.0
outputRecordType	String	Type of amendment record that you want to create.	Required	62.0
quantityChange	Double	Quantity to add to or reduce from the asset's existing quantity.	Required	62.0

Response body for POST

Amendment

Asset Cancellation (POST)

Initiate the cancellation of an asset.

Special Access Rules

To use this API, you need the InitiateCancellation API permission set.

Resource

```
/connect/revenue-management/assets/actions/cancel
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/revenue-management/assets/actions/cancel
```

Available version

62.0

HTTP methods

POST

Request body for POST

JSON example

```
{
  "assetIds": [
    "02iSG0000003NMhYAM",
    "02iSG0000006DvSYAU"
  ],
  "cancellationDate": "2023-10-04T00:00:00",
  "contractId": "800SG00000CFpepYAD",
  "opportunityId": "006SG000004W5tVYAS",
  "outputRecordId": "801SG00000DX1jWYAT",
  "outputRecordType": "Quote"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
assetIds	String[]	IDs of the assets that you want to cancel.	Required	62.0
cancellationDate	String	Effective date of the cancellation.	Required	62.0
contractId	String	ID of the Contract record that you want to sync with the cancellation quote.	Optional	62.0
opportunityId	String	ID of the Opportunity record that you want to sync with the cancellation quote.	Optional	62.0
outputRecordId	String	ID of the quote or order that you want to cancel.	Optional	62.0
outputRecordType	String	Type of cancellation record that you want to create.	Required	62.0

Response body for POST

Cancellation

Asset Renewal (POST)

Initiate the renewal of an asset.

Special Access Rules

To use this API, you need the InitiateRenewal API permission set.

Resource

```
/connect/revenue-management/assets/actions/renew
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/revenue-management/assets/actions/renew
```

Available version

62.0

HTTP methods

POST

Request body for POST

JSON example

```
{
  "assetIds": [
    "02iSG0000003NMhYAM",
    "02iSG0000006DvSYAU"
  ],
  "contractId": "800SG00000CFpepYAD",
  "opportunityId": "006SG000004W5tVYAS",
  "outputRecordId": "801SG00000DX1jWYAT",
  "outputRecordType": "Quote",
  "renewalEndDate": "2024-10-03T23:59:59",
  "renewalStartDate": "2023-10-04T00:00:00"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
assetIds	String[]	IDs of the assets that you want to renew.	Required	62.0
contractId	String	ID of the Contract record that you want to sync with the renewal of the Quote or Order record.	Optional	62.0
opportunityId	String	ID of the Opportunity record that you want to sync with the renewal quote.	Optional	62.0
outputRecordId	String	ID of the Quote or Order record that you want to renew.	Optional	62.0
outputRecordType	String	Type of renewal record that you want to create.	Required	62.0
renewalEndDate	String	End date of the renewal process for the assets.	Optional	62.0
renewalStartDate	String	Start date of the renewal process for the assets.	Optional	62.0

Response body for POST

Renewal

Request Bodies

Learn more about the available request bodies of Asset Lifecycle Business APIs.

Amendment Input

Input representation of the details of the request to create an amendment record.

Cancellation Input

Input representation of the details of the request to cancel a quote or an order.

Renewal Input

Input representation of the details of the request to initiate the renewal of an asset.

Amendment Input

Input representation of the details of the request to create an amendment record.

JSON example

```
{
  "assetIds": [
    "02iSG0000003NMhYAM",
    "02iSG0000006DvSYAU"
  ],
  "amendmentStartDate": "2023-10-04T00:00:00",
  "contractId": "800SG00000CFpepYAD",
  "opportunityId": "006SG000004W5tVYAS",
  "outputRecordId": "801SG00000DX1jWYAT",
  "outputRecordType": "Quote",
  "quantityChange": 5
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
assetIds	String[]	IDs of the assets that you want to add to the amendment record.	Required	62.0
amendmentStartDate	String	Start date of the amendment.	Required	62.0
contractId	String	ID of the Contract record that you want to sync with the amendment quote.	Optional	62.0
opportunityId	String	ID of the Opportunity record that you want to sync with the amendment quote.	Optional	62.0
outputRecordId	String	ID of the quote or order record that you want to add the assets to.	Optional	62.0
outputRecordType	String	Type of amendment record that you want to create.	Required	62.0
quantityChange	Double	Quantity to add to or reduce from the asset's existing quantity.	Required	62.0

Cancellation Input

Input representation of the details of the request to cancel a quote or an order.

JSON example

```
{
  "assetIds": [
    "02iSG0000003NMhYAM",
    "02iSG0000006DvSYAU"
  ],
  "cancellationDate": "2023-10-04T00:00:00",
  "contractId": "800SG00000CFpepYAD",
  "opportunityId": "006SG000004W5tVYAS",
  "outputRecordId": "801SG00000DX1jWYAT",
  "outputRecordType": "Quote"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
assetIds	String[]	IDs of the assets that you want to cancel.	Required	62.0
cancellationDate	String	Effective date of the cancellation.	Required	62.0
contractId	String	ID of the Contract record that you want to sync with the cancellation quote.	Optional	62.0
opportunityId	String	ID of the Opportunity record that you want to sync with the cancellation quote.	Optional	62.0
outputRecordId	String	ID of the quote or order that you want to cancel.	Optional	62.0
outputRecordType	String	Type of cancellation record that you want to create.	Required	62.0

Renewal Input

Input representation of the details of the request to initiate the renewal of an asset.

JSON example

```
{
  "assetIds": [
    "02iSG0000003NMhYAM",
    "02iSG0000006DvSYAU"
  ],
  "contractId": "800SG00000CFpepYAD",
  "opportunityId": "006SG000004W5tVYAS",
  "outputRecordId": "801SG00000DX1jWYAT",
  "outputRecordType": "Quote",
  "renewalEndDate": "2024-10-03T23:59:59",
```

```
  "renewalStartDate": "2023-10-04T00:00:00"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
assetIds	String[]	IDs of the assets that you want to renew.	Required	62.0
contractId	String	ID of the Contract record that you want to sync with the renewal of the Quote or Order record.	Optional	62.0
opportunityId	String	ID of the Opportunity record that you want to sync with the renewal quote.	Optional	62.0
output RecordId	String	ID of the Quote or Order record that you want to renew.	Optional	62.0
output RecordType	String	Type of renewal record that you want to create.	Required	62.0
renewal EndDate	String	End date of the renewal process for the assets.	Optional	62.0
renewal StartDate	String	Start date of the renewal process for the assets.	Optional	62.0

Response Bodies

Learn more about the available response bodies of Asset Lifecycle Business APIs.

Amendment
Output representation of the details of an amendment record.

ARC Base Error
Output representation of the error response related to the amendment, renewal, or cancellation of assets.

Cancellation
Output representation of the details of a cancellation record.

Renewal
Output representation of the details of a renewal record.

Amendment

Output representation of the details of an amendment record.

JSON example

```
{
  "amendmenRecordId": "0Q0xx0000004NsSCAU",
  "errors": [
```

```
    {
      "errorCode": "REQUIRED_FIELD_MISSING",
      "errorMessage": "Specify a value for quantityChange, and try again."
    }
  ],
  "requestId": "16Pxx0000004NIy",
  "success": true
}
```

Properties

Property Name	Type	Description	Filter Group and Version	Available Version
amendmentRecordId	String	ID of the amendment record that's created for a quote or an order.	Small, 62.0	62.0
errors	ARC Base Error on page 1011[]	Error responses if the creation of an amendment record fails.	Small, 62.0	62.0
requestId	String	Request ID that's used to track an async request.	Small, 62.0	62.0
success	Boolean	Indicates whether the API request is successful (true) or not (false).	Small, 62.0	62.0

ARC Base Error

Output representation of the error response related to the amendment, renewal, or cancellation of assets.

```
  "errors": [
      {
          "errorCode": "REQUIRED_FIELD_MISSING",
          "errorMessage": "Specify a value for quantityChange, and try again."
      }
  ]
```

Property Name	Type	Description	Filter Group and Version	Available Version
errorCode	String	Code for the resultant error.	Big, 62.0	62.0
errorMessage	String	Error message for the resultant error.	Big, 62.0	62.0

Cancellation

Output representation of the details of a cancellation record.

JSON example

```
{
  "cancellationRecordId": "0Q0xx0000004NsSCAU",
  "errors": [
    {
```

```
      "errorCode": "REQUIRED_FIELD_MISSING",
      "errorMessage": "Specify a value for quantityChange, and try again."
    }
  ],
  "requestId": "16Pxx0000004NIy",
  "success": true
}
```

Properties

Property Name	Type	Description	Filter Group and Version	Available Version
`cancellationRecordId`	String	ID of the cancellation record that's created for a quote or an order.	Small, 62.0	62.0
`errors`	ARC Base Error on page 1011[]	Error responses if the creation of a cancellation record fails.	Small, 62.0	62.0
`requestId`	String	Request ID that's used to track the async request.	Small, 62.0	62.0
`success`	Boolean	Indicates whether the API request is successful (`true`) or not (`false`).	Small, 62.0	62.0

Renewal

Output representation of the details of a renewal record.

JSON example

```
{
  "renewalRecordId": "0Q0xx0000004NsSCAU",
  "errors": [
    {
      "errorCode": "REQUIRED_FIELD_MISSING",
      "errorMessage": "Specify a value for quantityChange, and try again."
    }
  ],
  "requestId": "16Pxx0000004NIy",
  "success": true
}
```

Properties

Property Name	Type	Description	Filter Group and Version	Available Version
`errors`	ARC Base Error on page 1011[]	Error responses if the creation of a renewal record fails.	Small, 62.0	62.0
`renewalRecordId`	String	ID of the renewal record that's created for the Quote or Order record.	Small, 62.0	62.0

Property Name	Type	Description	Filter Group and Version	Available Version
`requestId`	String	Request ID that's used to track the async request.	Small, 62.0	62.0
`success`	Boolean	Indicates whether the API request is successful (`true`) or not (`false`).	Small, 62.0	62.0

CHAPTER 6 Usage Management

In this chapter ...

- Usage Management Standard Objects
- Usage Management Standard Invocable Actions
- Usage Management Business APIs
- Usage Management Metadata API Types

Provide transparent, accurate, and efficient management of usage data and estimated usage amount to enhance revenue management capabilities.

EDITIONS

Available in: Lightning Experience

Available in: **Enterprise**, **Unlimited**, and **Developer** Editions

Usage Management Standard Objects

The Usage Management data model provides objects and fields to set up and manage consumption of usage-based products.

ProductUsageGrant

Represents the details of a grant associated with a resource, product, or service, such as the purchased quantity, renewal and rollover policy, and validity of the grant. This object is available in API version 62.0 and later.

TransactionJournal

Represents consumption details of a usage resource that are recorded for creating usage summaries. This object is available for usage management in API version 63.0 and later.

TransactionUsageEntitlement

Represents the details of each usage entitlement that's granted with the purchased sellable product, such as quantity and date when the entitlements were granted. This object is available in API version 63.0 and later.

UnitOfMeasure

Defines the units and systems of units used to account for quantities of a usage resource. This object is available for usage management in API version 62.0 and later.

UnitOfMeasureClass

Represents a standard unit of measure dimension. This object is available in API version 63.0 and later.

UsageBillingPeriodItem

Represents the calculated overages for the usage entitlement and the amount that's charged for these overages. This object is available in API version 63.0 and later.

UsageEntitlementAccount

Represents the entitlement account details related to the asset that holds the wallet with the granted units. This object is available in API version 63.0 and later.

UsageEntitlementBucket

Represents a usage entitlement that's granted with the sellable product. This object is available in API version 63.0 and later.

UsageEntitlementEntry

Represents the usage entitlement details, such as the usage consumption, rollovers, and details of expired units for each tenure. This object is available in API version 63.0 and later.

UsageGrantRenewalPolicy

Represents a policy about the rollover of a usage grant. This object is available in API version 62.0 and later.

UsageGrantRolloverPolicy

Represents a policy about the rollover of a usage grant.This object is available in API version 62.0 and later.

UsagePrdGrantBindingPolicy

Represents the association of a usage resource's grants with a sellable product. This object is available in API version 63.0 and later.

UsageRatableSummary

Represents the aggregation of the usage summaries that are used to calculate the rate at which the overages are charged. This object is available in API version 63.0 and later.

UsageResource

Represents an entitlement granted to a user or party by a provider, such as data storage, computing power, bandwidth, or any other product or service. Additionally, this object is used to represent resources consumed over time. This object is available in API version 62.0 and later.

UsageResourceBillingPolicy

Represents information about how the usage is accumulated before rating a usage resource.This object is available in API version 62.0 and later.

UsageSummary

Represents the aggregation of the entries in the transaction journal for a usage entitlement for a specified period. This object is available in API version 63.0 and later.

SEE ALSO:

Object Reference for the Salesforce Platform: Overview of Salesforce Objects and Fields

SOAP API Developer Guide: Introduction to SOAP API

ProductUsageGrant

Represents the details of a grant associated with a resource, product, or service, such as the purchased quantity, renewal and rollover policy, and validity of the grant. This object is available in API version 62.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

To create, update, and delete product usage grant records, you must have the Usage Management Design Time permission set license.

Fields

Field	Details
`DrawdownOrder`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The order that's used to debit entitlement consumption from the usage entitlement bucket. Valid values are: • `ExpiringFirst` • `GrantedFirst` • `GrantedLast`
`EffectiveEndDate`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update

Field	Details
	Description The date and time until when the grant remains effective.
`EffectiveStartDate`	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** The date and time when the grant becomes effective.
`Label`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The identifying label for the product usage grant.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date when this record was last referenced.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed a record related to this record.
`OverageChargeable`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Specifies whether to charge for overages. Valid values are: • `No` • `Yes`

Field	Details
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The ID of the owner of the product usage grant. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
ProductOfferId	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The sellable product that grants the usage resource. This field is a relationship field. **Relationship Name** ProductOffer **Refers To** Product2
ProductUsageGrantNum	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The number of each resource grant map that starts with one and is consecutive.
Quantity	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The quantity of the granted resource.
RenewalPolicyId	**Type** reference

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update **Description** The grant renewal policy associated with the product usage grant. This field is a relationship field. **Relationship Name** RenewalPolicy **Refers To** UsageGrantRenewalPolicy
`RolloverPolicyId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The grant rollover policy associated with the product usage grant. This field is a relationship field. **Relationship Name** RolloverPolicy **Refers To** UsageGrantRolloverPolicy
`Status`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The status of the product usage grant. Valid values are: • `Active` • `Draft` • `Inactive`
`UnitOfMeasureClassId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The unit of measure class associated with the product usage grant. This field is a relationship field.

Field	Details
	Relationship Name UnitOfMeasureClass **Refers To** UnitOfMeasureClass
`UnitOfMeasureId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The unit for measure associated with the product usage grant. This value when specified, overrides the default unit of measure defined in the associated unit of measure class. This field is a relationship field. **Relationship Name** UnitOfMeasure **Refers To** UnitOfMeasure
`UsageDefinitionProductId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The sellable product associated with the usage resource that's used to retrieve tax policy, calculate rating during overages, and other invoicing actions. This field is a relationship field. **Relationship Name** UsageDefinitionProduct **Refers To** Product2
`UsageResourceId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The usage resource associated with the product usage grant. This field is a relationship field. **Relationship Name** UsageResource

Field	Details
	Refers To UsageResource
`ValidityPeriodTerm`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The period until when the resource grant is valid.
`ValidityPeriodUnit`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The length of a validity period for the resource grant, when used with the `ValidityPeriodTerm` field. Valid values are: • `Month` • `None` • `Year`

Associated Objects

This object has these associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ProductUsageGrantHistory
History is available for tracked fields of the object.

ProductUsageGrantOwnerSharingRule
Sharing rules are available for the object.

ProductUsageGrantShare
Sharing is available for the object.

TransactionJournal

Represents consumption details of a usage resource that are recorded for creating usage summaries. This object is available for usage management in API version 63.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`AccountId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The account associated with the transaction. This field is a relationship field. **Relationship Name** Account **Refers To** Account
`ActivityDate`	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** The date of the transaction.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date when this record was last referenced.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed a record related to this record.
`Name`	**Type** string

Field	Details
	Properties Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The autogenerated number of the transaction journal.
`ReferenceRecordId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The record of an asset that's associated with this transaction. This field is a polymorphic relationship field. **Relationship Name** ReferenceRecord **Refers To** Asset
`Status`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update **Description** The status of the transaction. Valid values are: • `Approved` • `Cancelled` • `Error` • `Pending` • `Processed` • `ProcessingError` • `Rejected` • `SystemApproved` • `SystemRejected` The default value is `Pending`.
`UniqueIdentifier`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description An autogenerated identifier for the transaction journal when the usage type is Usage Management.
`UsageResourceId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The usage resource that's associated with the transaction journal. This field is a relationship field. **Relationship Name** UsageResource **Refers To** UsageResource
`UsageType`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort **Description** The type of usage. Valid value is: • `UsageManagement`

Associated Objects

This object has these associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

TransactionJournalChangeEvent
Change events are available for the object.

TransactionUsageEntitlement

Represents the details of each usage entitlement that's granted with the purchased sellable product, such as quantity and date when the entitlements were granted. This object is available in API version 63.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

To create, update, and delete transaction usage entitlement records, you must have the Usage Management Run Time permission set license.

Fields

Field	Details
`AccountId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The account that's associated with the usage entitlement. This field is a relationship field. **Relationship Name** Account **Refers To** Account
`ActionType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The type of action that resulted in the transaction usage entitlement. Valid values are: • `Amend` • `Cancellation` • `New` • `Ramp` • `Renewal`
`AssetId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The asset associated with the sellable product. This field is a relationship field. **Relationship Name** Asset

Field	Details
	Refers To Asset
`ChargeForOverage`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The action to be taken when the entitlements are used beyond their grant values. Valid values are: • `No`—Don't charge for over consumption • `Yes`—Charge for over consumption
`DrawdownOrder`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The order that's used to debit entitlement consumption from the usage entitlement bucket. Valid values are: • `ExpiringFirst` • `GrantedFirst` • `GrantedLast`
`EffectiveEndDateTime`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date and time when the active transaction usage entitlement ends.
`EffectiveStartDateTime`	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** The date and time when the transaction usage entitlement becomes active.
`EntitlementUomClassId`	**Type** reference

Field	Details
	Properties Filter, Group, Nillable, Sort **Description** The unit of measure class for the usage entitlement. This field is a relationship field. **Relationship Name** EntitlementUomClass **Refers To** UnitOfMeasureClass
`EntitlementUomId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The unit of measure to calculate the usage entitlement. This field is a relationship field. **Relationship Name** EntitlementUom **Refers To** UnitOfMeasure
`ExternalGrantBindingTarget`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The custom target associated with the entitlements that are granted with the sellable product.
`ExternalOrderItem`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The custom or external order item that's associated with the entitlement.
`GrantBindingTargetId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The target associated with the entitlements that are granted with the sellable product. This field is a relationship field. **Relationship Name** GrantBindingTarget **Refers To** Asset
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date when this record was last referenced.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed a record related to this record.
Name	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Autogenerated identifier for the transaction usage entitlement record.
NetQuantity	**Type** double **Properties** Create, Filter, Sort, Update **Description** The total quantity that combines the amended quantity with the initial transaction quantity in the order item.
OrderItemId	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The order item that's associated with the entitlement. This field is a polymorphic relationship field. **Relationship Name** OrderItem **Refers To** OrderItem, WorkOrderLineItem
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The ID of the owner of the transaction usage entitlement. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
`PricebookEntryId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The price book entry that's associated with the sellable product. This field is a relationship field. **Relationship Name** PricebookEntry **Refers To** PricebookEntry
`ProductId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The sellable product for which the entitlement is granted. This field is a relationship field.

Field	Details
	Relationship Name Product **Refers To** Product2
`EntitlementQuantity`	**Type** double **Properties** Create, Filter, Sort, Update **Description** The entitlement quantity for the usage resource.
`TransactionQuantity`	**Type** double **Properties** Create, Filter, Sort, Update **Description** The transaction quantity in the order for the usage entitlement.
`UsageAggregationPolicyId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The usage aggregation policy for this entitlement. This field is a relationship field. **Relationship Name** UsageAggregationPolicy **Refers To** UsageResourceBillingPolicy
`UsageGrantRefreshPolicyId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The usage grant refresh policy that's associated with the transaction usage entitlement. This field is a relationship field. **Relationship Name** UsageGrantRefreshPolicy

Field	Details
	Refers To UsageGrantRenewalPolicy
`UsageGrantRolloverPolicyId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The usage grant rollover policy that's associated with the transaction usage entitlement. This field is a relationship field. **Relationship Name** UsageGrantRolloverPolicy **Refers To** UsageGrantRolloverPolicy
`UsageResourceId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The usage resource record that's associated with the transaction usage entitlement. This field is a relationship field. **Relationship Name** UsageResource **Refers To** UsageResource
`ValidityPeriodTerm`	**Type** int **Properties** Create, Filter, Group, Sort, Update **Description** The duration for which the usage resource grant is valid, when used with the validity period units.
`ValidityPeriodUnit`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update

Field	Details
	Description The length of a validity period for the usage resource grant, when used with the validity period term. Valid values are: • `Month` • `None` • `Year`

Associated Objects

This object has these associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

TransactionUsageEntitlementHistory
History is available for tracked fields of the object.

TransactionUsageEntitlementOwnerSharingRule
Sharing rules are available for the object.

TransactionUsageEntitlementShare
Sharing is available for the object.

UnitOfMeasure

Defines the units and systems of units used to account for quantities of a usage resource. This object is available for usage management in API version 62.0 and later.

Examples of units of measure include Liter (for volume), Kilogram (for weight), and single units (such as Can, sachet, and packet).

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`ConversionFactor`	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The factor or rate that's used to convert this unit of measurement to the base unit.

Field	Details
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date when this record was last referenced.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed a record related to this record.
Name	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** Autogenerated identifier for the unit of measure record.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The ID of the owner of the unit of measure record. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
Sequence	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The sequence number assigned to the unit of measure.

Field	Details
`Status`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The status of the unit of measure. Valid values are: • `Active` • `Draft` • `Inactive`
`Type`	**Type** picklist **Properties** Create, Filter, Group, Sort, Update **Description** The type of the unit of measure. For example, weight, distance, period.
`UnitCode`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** The code for the unit of measure.
`UnitOfMeasureClassId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The unit of measure class associated with the unit of measure. This field is a relationship field. **Relationship Name** UnitOfMeasureClass **Refers To** UnitOfMeasureClass

Associated Objects

This object has these associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

UnitOfMeasureChangeEvent
Change events are available for the object.

UnitOfMeasureShare
Sharing is available for the object.

UnitOfMeasureClass

Represents a standard unit of measure dimension. This object is available in API version 63.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`BaseUnitOfMeasureId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The base unit of measure that's used to calculate and display quantities. This field is a relationship field. **Relationship Name** BaseUnitOfMeasure **Refers To** UnitOfMeasure
`Code`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** The unique user-defined string for the unit of measure class.
`DefaultUnitOfMeasureId`	**Type** reference

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update **Description** The default unit of measure that's used to calculate and display quantities. This measure can be different from the base unit of measure and overridden by individual resources. This field is a relationship field. **Relationship Name** DefaultUnitOfMeasure **Refers To** UnitOfMeasure
`Description`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The description of the unit of measure class.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date when this record was last referenced.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed a record related to this record.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** Autogenerated identifier for the unit of measure class record.
`Status`	**Type** picklist

Field	Details
	Properties Create, Filter, Group, Restricted picklist, Sort, Update **Description** The status of the unit of measure class. Valid values are: • `Active` • `Draft` • `Inactive`

Associated Objects

This object has these associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

UnitOfMeasureClassHistory
History is available for tracked fields of the object.

UsageBillingPeriodItem

Represents the calculated overages for the usage entitlement and the amount that's charged for these overages. This object is available in API version 63.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

To create, update, and delete transaction usage entitlement records, you must have the Usage Management Run Time permission set license.

Fields

Field	Details
`AccountId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The account that's associated with the billing period item. This field is a relationship field.

Field	Details
	Relationship Name Account **Refers To** Account
AssetId	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The asset that's associated with the usage billing period item. This field is a relationship field. **Relationship Name** Asset **Refers To** Asset
EndDateTime	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** The end date and time of the billing period.
ErrorCode	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The error code when the billing request fails. Valid value is: • INTERNAL_ERROR
ErrorDescription	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The error description that corresponds to the error code.

Field	Details
ExternalGrantBindingTarget	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The custom record that's associated with the entitlements that are granted with the sellable product.
GrantBindingTargetId	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The record that's associated with the usage entitlement account. This field is a relationship field. **Relationship Name** GrantBindingTarget **Refers To** Asset
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date when this record was last referenced.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed a record related to this record.
OverageAmount	**Type** currency **Properties** Create, Filter, Sort, Update **Description** The amount calculated for the overage.

Field	Details
`OverageAmountDerived`	**Type** double **Properties** Filter, Nillable, Sort **Description** The numeric value specified in the `OverageAmount` field to process Data processing Engine (DPE) jobs. This field is a calculated field.
`OverageQuantity`	**Type** double **Properties** Create, Filter, Sort, Update **Description** The quantity of the usage entitlement that was overused for the specified billing period.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The ID of the owner of the usage billing period item. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
`StartDateTime`	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** The start date and time of the billing period.
`Status`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The status of the usage billing period item.

Field	Details
	Valid values are: • `Invoiced` • `InvoicingFailed` • `LiableSummaryComplete` • `New` • `ReadyForInvoicing`
`TotalUsedQuantity`	**Type** double **Properties** Create, Filter, Sort, Update **Description** The total quantity of usage entitlement that was used for the billing period. This includes granted and overused entitlements.
`UoMId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The unit of measure for the overage quantity of usage entitlement. This field is a relationship field. **Relationship Name** UoM **Refers To** UnitOfMeasure
`UsageBillingPeriodItemNum`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Autogenerated identifier for the usage billing period item record.
`UsageEntitlementAccountId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The usage entitlement account associated with the source usage entitlement bucket. This field is a relationship field.

Field	Details
	Relationship Name UsageEntitlementAccount **Refers To** UsageEntitlementAccount
`UsageEntitlementBucketId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The usage entitlement bucket that's associated with the billing period item. This field is a relationship field. **Relationship Name** UsageEntitlementBucket **Refers To** UsageEntitlementBucket
`UsageResourceId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The usage resource associated with the usage entitlement bucket. This field is a relationship field. **Relationship Name** UsageResource **Refers To** UsageResource

Associated Objects

This object has these associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

UsageBillingPeriodItemFeed
Feed tracking is available for the object.

UsageBillingPeriodItemHistory
History is available for tracked fields of the object.

UsageBillingPeriodItemShare
Sharing is available for the object.

UsageEntitlementAccount

Represents the entitlement account details related to the asset that holds the wallet with the granted units. This object is available in API version 63.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`AccountId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The account that's associated with the usage entitlement. This field is a relationship field. **Relationship Name** Account **Refers To** Account
`BillDayOfMonth`	**Type** int **Properties** Create, Filter, Group, Sort, Update **Description** The day of the month that the bill is generated on and that ends the billing period.
`BillingPeriodEndDateTime`	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** The date and time when the billing period of the usage entitlements ends.
`BillingPeriodStartDateTime`	**Type** dateTime **Properties** Create, Filter, Sort, Update

Field	Details
	Description The date and time when the billing period of the usage entitlements starts.
`BillingPeriodTerm`	**Type** int **Properties** Create, Filter, Group, Sort, Update **Description** The frequency at which the bill is generated.
`BillingPeriodUnit`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The unit to measure the billing frequency. Valid value is: • `MONTH`
`EffectiveEndDateTime`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date and time when the usage entitlement account ends.
`EffectiveStartDateTime`	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** The date and time when the usage entitlement account starts.
`ExternalGrantBindingTarget`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The custom target that's associated with the entitlements that are granted with the sellable product.

Field	Details
GrantBindingTargetId	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The target that's associated with the usage entitlement account. This field is a relationship field. **Relationship Name** GrantBindingTarget **Refers To** Asset
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date when this record was last referenced.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed a record related to this record.
Name	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Autogenerated identifier for the usage entitlement account record.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The ID of the owner of the usage entitlement account. This field is a polymorphic relationship field.

Field	Details
	Relationship Name Owner **Refers To** Group, User
`PricebookEntryId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The price book entry that's associated with the sellable product. This field is a relationship field. **Relationship Name** PricebookEntry **Refers To** PricebookEntry
`ProductId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The sellable product for which the entitlements are granted. This field is a relationship field. **Relationship Name** Product **Refers To** Product2

Associated Objects

This object has these associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

UsageEntitlementAccountFeed
Feed tracking is available for the object.

UsageEntitlementAccountHistory
History is available for tracked fields of the object.

UsageEntitlementAccountShare
Sharing is available for the object.

UsageEntitlementBucket

Represents a usage entitlement that's granted with the sellable product. This object is available in API version 63.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`BucketBalance`	**Type** double **Properties** Create, Filter, Sort, Update **Description** The balance of the usage entitlement bucket after each transaction.
`BucketBalanceUomId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The unit that's used to measure the usage entitlement bucket balance. This field is a relationship field. **Relationship Name** BucketBalanceUom **Refers To** UnitOfMeasure
`CompletedRollovers`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The number of rollovers that were completed for the entitlements.
`ConsumedEntitlement`	**Type** double **Properties** Create, Filter, Nillable, Sort, Update

Field	Details
	Description The entitlements that have been consumed from this bucket.
`EffectiveEndDateTime`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date and time when the usage entitlement bucket period ends.
`EffectiveStartDateTime`	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** The date and time when the usage entitlement bucket becomes effective.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date when this record was last referenced.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed a record related to this record.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Autogenerated identifier for the usage entitlement bucket record.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update

Field	Details
	Description The ID of the owner of the usage entitlement bucket. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
`ParentId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The parent record that's associated with the usage entitlement bucket. This field is a polymorphic relationship field. **Relationship Name** Parent **Refers To** UsageEntitlementAccount, UsageEntitlementBucket
`TotalAsOfBalance`	**Type** double **Properties** Filter, Nillable, Sort **Description** The balance of the entitlements granted with the usage resource for a Usage entitlement account.
`TotalConsumedEntitlement`	**Type** double **Properties** Filter, Nillable, Sort **Description** The entitlements that have been consumed from this bucket.
`TransactionUsageEntitlementId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The transaction usage entitlement associated with this usage entitlement bucket.

Field	Details
	This field is a relationship field. **Relationship Name** TransactionUsageEntitlement **Refers To** TransactionUsageEntitlement
`UsageResourceId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The usage resource that's associated with the usage entitlement bucket. This field is a relationship field. **Relationship Name** UsageResource **Refers To** UsageResource

Associated Objects

This object has these associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

UsageEntitlementBucketFeed
Feed tracking is available for the object.

UsageEntitlementBucketHistory
History is available for tracked fields of the object.

UsageEntitlementBucketShare
Sharing is available for the object.

UsageEntitlementEntry

Represents the usage entitlement details, such as the usage consumption, rollovers, and details of expired units for each tenure. This object is available in API version 63.0 and later.

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.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`ClosingBalance`	**Type** double **Properties** Create, Filter, Sort, Update **Description** The quantity of the entitlement after adjusting the transaction quantity for the current entitlement validity period.
`EffectiveEndDateTime`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date and time when the usage entitlement entry ends.
`EffectiveStartDateTime`	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** The date and time when the usage entitlement entry starts.
`EntitlementBucketId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The usage entitlement bucket that's associated with the entitlement entry. This field is a relationship field. **Relationship Name** EntitlementBucket **Relationship Type** Master-detail **Refers To** UsageEntitlementBucket (the master object)
`LastReferencedDate`	**Type** dateTime

Field	Details
	Properties Filter, Nillable, Sort **Description** The date when this record was last referenced.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed a record related to this record.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Autogenerated identifier for the usage entitlement entry record.
`OpeningBalance`	**Type** double **Properties** Create, Filter, Sort, Update **Description** The starting quantity of the entitlement for the current entitlement validity period.
`ParentEntitlementBucketId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The parent usage entitlement bucket record that's associated with the entitlement entry. This field is a relationship field. **Relationship Name** ParentEntitlementBucket **Refers To** UsageEntitlementBucket
`TransactionDate`	**Type** dateTime

Field	Details
	Properties Create, Filter, Sort, Update **Description** The date when the transaction usage entitlement entry was updated.
`TransactionQuantity`	**Type** double **Properties** Create, Filter, Sort, Update **Description** The quantity of the entitlement used during the current entitlement validity period.
`TransactionReason`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The reason for the usage entitlement entry. Valid values are: • `Amend` • `Cancel` • `Consumption` • `Original` • `Refresh` • `Renew` • `Rollover`
`TransactionType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The type of transaction. Valid values are: • `Credit` • `Debit` • `Expired`
`TransactionUsageEntitlementId`	**Type** reference

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update **Description** The transaction usage entitlement that's associated with this usage entitlement entry. This field is a relationship field. **Relationship Name** TransactionUsageEntitlement **Refers To** TransactionUsageEntitlement
`TransactionalBucketId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The usage entitlement bucket that's used to debit the consumption of entitlements. This field is a relationship field. **Relationship Name** TransactionalBucket **Refers To** UsageEntitlementBucket
`UomId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The unit of measure that's associated with the usage entitlement. This field is a relationship field. **Relationship Name** Uom **Refers To** UnitOfMeasure

Associated Objects

This object has these associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

UsageEntitlementEntryFeed

Feed tracking is available for the object.

UsageEntitlementEntryHistory
History is available for tracked fields of the object.

UsageEntitlementEntryOwnerSharingRule
Sharing rules are available for the object.

UsageEntitlementEntryShare
Sharing is available for the object.

UsageGrantRenewalPolicy

Represents a policy about the rollover of a usage grant. This object is available in API version 62.0 and later.

A usage grant renewal policy is used if you want to never renew a usage grant or renew on a specific frequency.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Code`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** A unique user-defined string for the usage grant renewal policy.
`IsRenewalAllowed`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the policy renewal is allowed (`true`) or not (`false`). If `true`, then the policy can be renewed. The default value is `false`.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date when this record was last referenced.

Field	Details
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed a record related to this record.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** The name of the renewal policy record.
`RenewalFrequency`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The frequency of the policy renewals, when used with the `RenewalFrequencyUnit` field.
`RenewalFrequencyUnit`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The renewal duration for a policy. Valid values are: • `Month` • `Quarter` • `Year`
`Status`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The status of the renewal policy. Valid values are: • `Active`

Field	Details
	• `Draft` • `Inactive`

Associated Objects

This object has these associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

UsageGrantRenewalPolicyFeed
Feed tracking is available for the object.

UsageGrantRenewalPolicyHistory
History is available for tracked fields of the object.

UsageGrantRolloverPolicy

Represents a policy about the rollover of a usage grant.This object is available in API version 62.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Code`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** A unique user-defined string for the usage grant rollover policy.
`IsRolloverAllowed`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the policy allows the rollover of the usage grant. The default value is `false`.
`LastReferencedDate`	**Type** dateTime

Field	Details
	Properties Filter, Nillable, Sort **Description** The date when this record was last referenced.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed a record related to this record.
`MaximumRolloverCount`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The maximum number of times that the usage grant can roll over.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** The name of the rollover policy record.
`ShouldAllowRolloverExpiry`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the rollover for the associated usage grant is allowed to expire. The default value is `false`.
`Status`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Specifies the status of the rollover policy. Possible values are:

Field	Details
	• `Active` • `Draft` • `Inactive`

Associated Objects

This object has these associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

UsageGrantRolloverPolicyFeed
Feed tracking is available for the object.

UsageGrantRolloverPolicyHistory
History is available for tracked fields of the object.

UsagePrdGrantBindingPolicy

Represents the association of a usage resource's grants with a sellable product. This object is available in API version 63.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`GrantBindingTargetType`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The type of binding instance to which the grant is added. Valid values are: • `Custom` • `Product`
`GrantBindingType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update

Field	Details
	Description The type of binding that indicates where the grant is added. Valid values are: • `Self` • `Target`
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date when this record was last referenced.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed a record related to this record.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** Autogenerated identifier for the usage product grant binding policy.
`Product2Id`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The sellable product associated with the usage product grant binding policy. This field is a relationship field. **Relationship Name** Product2 **Refers To** Product2

Associated Objects

This object has these associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

UsagePrdGrantBindingPolicyFeed
Feed tracking is available for the object.

UsagePrdGrantBindingPolicyHistory
History is available for tracked fields of the object.

UsageRatableSummary

Represents the aggregation of the usage summaries that are used to calculate the rate at which the overages are charged. This object is available in API version 63.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`AccountId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The account that's associated with the ratable summary. This field is a relationship field. **Relationship Name** Account **Refers To** Account
`AssetId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The asset that's associated with the ratable summary. This field is a relationship field. **Relationship Name** Asset

Field	Details
	Refers To Asset
`EndDateTime`	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** The end date and time of the ratable period.
`ErrorCode`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The code that represents the error encountered when the ratable summary is rated. Valid value is: • `INTERNAL_ERROR`
`ErrorDescription`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The error description that corresponds to the error code.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date when this record was last referenced.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed a record related to this record.

Field	Details
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Autogenerated identifier for the usage ratable summary record.
`NetUnitRate`	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The rate that's used to calculate the amount for each unit of usage entitlement overage.
`NetUnitRateUomId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The unit of measure for the net unit rate applied to the overage. This field is a relationship field. **Relationship Name** NetUnitRateUom **Refers To** UnitOfMeasure
`OverageQuantity`	**Type** double **Properties** Create, Filter, Sort, Update **Description** The quantity of the usage entitlements that were overused.
`OverageQuantityUomId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The unit of measure that's used for overage quantity. This field is a relationship field.

Field	Details
	Relationship Name OverageQuantityUom **Refers To** UnitOfMeasure
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The ID of the owner of the usage ratable summary. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
`RatingDecisionDateTime`	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** The date and time when the decision about rating consumption is done.
`RatingExecutionIdentifier`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ID of the rating service execution that's used to retrieve waterfall logs.
`RatingRequestId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ID that invokes the rating service that's used to rate the usage ratable summary. This field is a relationship field. **Relationship Name** RatingRequest

Field	Details
	Refers To RatingRequest
`StartDateTime`	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** The start date and time of the ratable period.
`Status`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The status of the ratable summary. Valid values are: • `New` • `RatingComplete` • `RatingFailed` • `SummaryCreated`
`TierQuantity`	**Type** double **Properties** Create, Filter, Sort, Update **Description** The quantity that impacts the net unit rate of a usage entitlement.
`TierQuantityUomId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The unit of measure for the tier quantity. This field is a relationship field. **Relationship Name** TierQuantityUom **Refers To** UnitOfMeasure

Field	Details
TotalAmount	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The total amount after calculating the overage based on the net unit rate.
UsageEntitlementAccountId	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The usage entitlement account that's associated with the source of the ratable summary. This field is a relationship field. **Relationship Name** UsageEntitlementAccount **Refers To** UsageEntitlementAccount
UsageEntitlementBucketId	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The usage entitlement bucket that's associated with the source of the ratable summary. This field is a relationship field. **Relationship Name** UsageEntitlementBucket **Refers To** UsageEntitlementBucket
UsageResourceId	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The usage resource associated with the usage entitlement bucket. This field is a relationship field. **Relationship Name** UsageResource

Field	Details
	Refers To UsageResource

Associated Objects

This object has these associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

UsageRatableSummaryHistory
History is available for tracked fields of the object.

UsageRatableSummaryOwnerSharingRule
Sharing rules are available for the object.

UsageRatableSummaryShare
Sharing is available for the object.

UsageResource

Represents an entitlement granted to a user or party by a provider, such as data storage, computing power, bandwidth, or any other product or service. Additionally, this object is used to represent resources consumed over time. This object is available in API version 62.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Category`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The category of the usage resource that's used to organize and understand the product grant maps. Valid value is: • `Usage`
`Code`	**Type** string

Field	Details
	Properties Create, Filter, Group, idLookup, Sort, Update **Description** The unique user-defined string for the usage resource.
`DefaultUnitOfMeasureId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The default unit of measure for the given resource. The default value can be overridden with an alternate default unit of measure for a given resource. This field is a relationship field. **Relationship Name** DefaultUnitOfMeasure **Refers To** UnitOfMeasure
`Description`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The description of the usage resource.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date when this record was last referenced.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed a record related to this record.
`Name`	**Type** string

Field	Details
	Properties Create, Filter, Group, idLookup, Sort, Update **Description** The name of the usage resource record.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The ID of the owner of the usage resource record. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
`Status`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The status of the usage resource. Valid values are: • `Active` • `Draft` • `Inactive`
`UnitOfMeasureClassId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The unit of measure class that's used with the resource to define the units in which this resource is measured. This field is a relationship field. **Relationship Name** UnitOfMeasureClass **Refers To** UnitOfMeasureClass

Field	Details
`UsageDefinitionProductId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The product associated with the usage resource to retrieve tax policy, calculate rating during overages, and other invoicing actions. This field is a relationship field. **Relationship Name** UsageDefinitionProduct **Refers To** Product2
`UsageResourceBillingPolicyId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The usage resource billing policy that defines how the usage resource can be aggregated before it's sent for rating. This field is a relationship field. **Relationship Name** UsageResourceBillingPolicy **Refers To** UsageResourceBillingPolicy

Associated Objects

This object has these associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

UsageResourceFeed
Feed tracking is available for the object.

UsageResourceHistory
History is available for tracked fields of the object.

UsageResourceOwnerSharingRule
Sharing rules are available for the object.

UsageResourceShare
Sharing is available for the object.

UsageResourceBillingPolicy

Represents information about how the usage is accumulated before rating a usage resource.This object is available in API version 62.0 and later.

A usage resource billing policy object is used to configure the properties of usage resources related to how aggregation is performed on the usage records before rating. Usage resource billing policies are defined at the usage resource level and can be reused across multiple usage resources.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Code`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** A unique user-defined string for the usage resource billing policy.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date when this record was last referenced.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed a record related to this record.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** The name of the usage resource billing policy record.

Field	Details
`Status`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The status of the policy. Valid values are: • `Active` • `Draft` • `Inactive`
`UsageAccumulationMethod`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The method used to accumulate the usage. Valid values are: • `Peak` • `Sum`
`UsageAccumulationPeriod`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The duration for which the usage is accumulated. Valid values are: • `Daily` • `Monthly`

Associated Objects

This object has these associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

UsageResourceBillingPolicyFeed
Feed tracking is available for the object.

UsageResourceBillingPolicyHistory
History is available for tracked fields of the object.

UsageSummary

Represents the aggregation of the entries in the transaction journal for a usage entitlement for a specified period. This object is available in API version 63.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`AccountId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The account that's associated with the sellable product that was purchased. This field is a relationship field. **Relationship Name** Account **Refers To** Account
`AssetId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The asset that's associated with the entitlement that was summarized. This field is a relationship field. **Relationship Name** Asset **Refers To** Asset
`ConsumptionUnits`	**Type** double **Properties** Create, Filter, Sort, Update **Description** The quantity that was used by the usage entitlement bucket.

Field	Details
DebitedUnits	**Type** double **Properties** Create, Filter, Sort, Update **Description** The units that are debited from the associated usage entitlement bucket.
EndDateTime	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** The end date and time of the usage summary.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date when this record was last referenced.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp for when the current user last viewed a record related to this record.
LiableSummaryId	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The usage billing period item that's associated with this usage summary. This field is a relationship field. **Relationship Name** LiableSummary **Refers To** UsageBillingPeriodItem

Field	Details
Name	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Autogenerated identifier for the usage summary record.
OverageUnits	**Type** double **Properties** Create, Filter, Sort, Update **Description** The quantity that was overused by the usage entitlement bucket.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The ID of the owner of the usage summary. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
RatableSummaryId	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ratable summary that's associated with this usage summary. This field is a relationship field. **Relationship Name** RatableSummary **Refers To** UsageRatableSummary
StartDateTime	**Type** dateTime

Field	Details
	Properties Create, Filter, Sort, Update **Description** The start date and time of the usage summary.
`Status`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The status of the usage summary. Valid values are: • `DrawdownComplete` • `LiableSummaryComplete` • `New` • `RatableSummaryComplete` • `Rated` • `UsageSummaryComplete` • `UsageSummaryInProgress`
`UomId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The unit of measure for the usage resource. This value overrides the default unit of measure defined in the associated unit of measure class. This field is a relationship field. **Relationship Name** Uom **Refers To** UnitOfMeasure
`UsageEntitlementAccountId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The usage entitlement account that's associated with the usage summary. This field is a relationship field.

Field	Details
	Relationship Name UsageEntitlementAccount **Refers To** UsageEntitlementAccount
`UsageEntitlementBucketId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The usage entitlement bucket that's associated with the usage summary. This field is a relationship field. **Relationship Name** UsageEntitlementBucket **Refers To** UsageEntitlementBucket
`UsageResourceId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The usage resource that's associated with the source of the usage entitlement bucket. This field is a relationship field. **Relationship Name** UsageResource **Refers To** UsageResource

Associated Objects

This object has these associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

UsageSummaryHistory
History is available for tracked fields of the object.

UsageSummaryOwnerSharingRule
Sharing rules are available for the object.

UsageSummaryShare
Sharing is available for the object.

Usage Management Standard Invocable Actions

Learn more about the standard invocable actions available with Usage Management.

Invoke Summary Creation Action

Invoke the service that creates various summaries, such as usage, ratable, and liable summaries where the usage amount is zero. The service also checks and updates the billing period of the usage entitlement account if the billing period is expired.

Process Consumption Overages Action

Process consumption overages for the usage summary records with `SummaryComplete` status. This action uses the entitlement service to process the overages.

Refresh Usage Entitlement Bucket Action

Refresh entitlements by evaluating the usage entitlement bucket records and creating a new usage entitlement entry.

SEE ALSO:

Actions Developer Guide: Overview

REST API Developer Guide: Invocable Actions Standard

Salesforce Help: Usage Management

Invoke Summary Creation Action

Invoke the service that creates various summaries, such as usage, ratable, and liable summaries where the usage amount is zero. The service also checks and updates the billing period of the usage entitlement account if the billing period is expired.

This action is available in API version 63.0 and later.

Special Access Rules

The Invoke Summary Creation action is available in Enterprise, Developer, and Unlimited Editions where Usage Management is enabled. To use this action, you need the Usage Management Run Time User permission.

Supported REST HTTP Methods

URI

`/services/data/v63.0/actions/standard/invokeSummaryCreationService`

Formats

JSON, XML

HTTP Methods

POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`usageEntitlementAccountId`	**Type** string **Description** Required. ID of the usage entitlement account record that's used to create summaries.

Outputs

None.

Example

POST

This example shows a sample request for the Invoke Summary Creation action.

```
{
  "inputs": [
    {
      "usageEntitlementAccountId": "3ttDU00000000iZYAQ"
    }
  ]
}
```

This example shows a sample response for the Invoke Summary Creation action.

```
{
  "actionName": "invokeSummaryCreationService",
  "errors": null,
  "isSuccess": true
}
```

SEE ALSO:

Salesforce Help: Permission Set Licenses, Personas, and User Permissions for Usage Management

Process Consumption Overages Action

Process consumption overages for the usage summary records with `SummaryComplete` status. This action uses the entitlement service to process the overages.

This action is available in API version 63.0 and later.

Special Access Rules

The Process Consumption Overages action is available in Enterprise, Developer, and Unlimited Editions where Usage Management is enabled. To use this action, you need the Usage Management Run Time User permission.

Supported REST HTTP Methods

URI

`/services/data/v63.0/actions/standard/processConsumptionOverages`

Formats

JSON, XML

HTTP Methods

POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`usageRatableSummaryId`	**Type** string **Description** Required. ID of the usage ratable summary record that contains the consumption details, and is used to calculate consumption overages and create usage entitlement entry records.

Outputs

None.

Example

POST

This example shows a sample request for the Process Consumption Overages action.

```
{
  "inputs": [
    {
      "usageRatableSummaryId": "3ttDU00000000iZYAQ"
    }
  ]
}
```

This example shows a sample response for the Process Consumption Overages action.

```
{
  "actionName": "processConsumptionOverages",
  "errors": null,
  "isSuccess": true
}
```

SEE ALSO:

Salesforce Help: Permission Set Licenses, Personas, and User Permissions for Usage Management

Refresh Usage Entitlement Bucket Action

Refresh entitlements by evaluating the usage entitlement bucket records and creating a new usage entitlement entry.

This action is available in API version 63.0 and later.

Special Access Rules

The Refresh Usage Entitlement Bucket action is available in Enterprise, Developer, and Unlimited Editions where Usage Management is enabled. To use this action, you need the Usage Management Run Time User permission.

Supported REST HTTP Methods

URI

`/services/data/v63.0/actions/standard/refreshUsageEntitlementBucket`

Formats

JSON, XML

HTTP Methods

POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`transactionUsageEntitlementId`	**Type** string **Description** Required. ID of the transaction usage entitlement record that's associated with the usage entitlement buckets that you need to refresh.

Outputs

None.

Example

POST

This example shows a sample request for the Refresh Usage Entitlement Bucket action.

```
{
  "inputs": [
    {
      "transactionUsageEntitlementId": "3ttDU00000000iZYAQ"
    }
  ]
}
```

This example shows a sample response for the Refresh Usage Entitlement Bucket action.

```
{
  "actionName": "refreshUsageEntitlementBucket",
  "errors": null,
  "isSuccess": true
}
```

SEE ALSO:

Salesforce Help: Permission Set Licenses, Personas, and User Permissions for Usage Management

Usage Management Business APIs

Use the Usage Management Business APIs to get details of a usage-based product that's associated with an asset, an order item, or a quote line item.

This table lists the available Usage Management resources.

Resource	Description
`/asset-management/assets/`***`assetId`***`/usage-details` (GET)	Get details of a usage-based product associated with an asset. This covers details of grants, resources, and configured rates for the product, including negotiated rates in case of a rate override.
`/commerce/sales-orders/line-items/`***`orderItemId`***`/usage-details` (GET)	Get details of a usage-based product associated with an order item.

Resource	Description
`/commerce/quotes/line-items/quoteLineItemId/usage-details` (GET)	Get details of a usage-based product associated with a quote line item.

Resources
Learn more about the available Usage Management API resources.

Response Bodies
Learn more about the available response bodies of Usage Management APIs.

SEE ALSO:

Connect REST API Developer Guide: Introduction

Resources

Learn more about the available Usage Management API resources.

Asset Usage Details (GET)
Get details of a usage-based product associated with an asset. This covers details of grants, resources, and configured rates for the product, including negotiated rates in case of a rate override.

Order Item Usage Details (GET)
Get details of a usage-based product associated with an order item.

Quote Line Item Usage Details (GET)
Get details of a usage-based product associated with a quote line item.

Asset Usage Details (GET)

Get details of a usage-based product associated with an asset. This covers details of grants, resources, and configured rates for the product, including negotiated rates in case of a rate override.

Here are the details of the information that this API returns:

- Grants and resources for the product, if rates aren't configured.
- Grants, resources, and any configured rates for the product. The rates are returned by the Rate Plan (GET) API.
- Resources that include grants, if applicable, and any negotiated rates for the product in case of a rate override request.

Resource

```
/asset-management/assets/assetId/usage-details
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/asset-management/assets/02iRM0000000tCdYAI/usage-details
```

Available version

63.0

HTTP methods

GET

Path parameter for GET

Parameter Name	Type	Description	Required or Optional	Available Version
`assetId`	String	ID of the asset.	Required	63.0

Query parameters for GET

Parameter Name	Type	Description	Required or Optional	Available Version
`effectiveDate`	String	Date that's used to search for the applicable rate card entries.	Optional	63.0
`optionalFields`	String[]	Custom fields that you can use to query these objects. • AssetRateCardEntry • AssetRateAdjustment	Optional	63.0

Response body for GET

Usage Details

Order Item Usage Details (GET)

Get details of a usage-based product associated with an order item.

Here are the details of the information that this API returns:

- Grants and resources for the product, if rates aren't configured.
- Grants, resources, and any configured rates for the product. The rates are returned by the Rate Plan (GET) API.
- Resources that include grants, if applicable, and any negotiated rates for the product in case of a rate override request.

Resource

```
/commerce/sales-orders/line-items/orderItemId/usage-details
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/commerce/sales-orders/line-items/802SG000003vZ15YAE/usage-details
```

```
https://yourInstance.salesforce.com/services/data
/v63.0/commerce/sales-orders/line-items/802SG000003vZ15YAE/usage-details?optionalFields=OrderItemRateCardEntryMCustomDate__c,OrderItemRateCardEntryMCustomNumber__c,OrderItemRateAdjustmentMCustomString__c
```

Available version

63.0

HTTP methods

GET

Path parameter for GET

Parameter Name	Type	Description	Required or Optional	Available Version
`orderItemId`	String	ID of the order item.	Required	63.0

Query parameters for GET

Parameter Name	Type	Description	Required or Optional	Available Version
`effectiveDate`	String	Date that's used to search for the applicable rate card entries.	Optional	63.0
`optionalFields`	String[]	Custom fields that you can use to query these objects. • OrderItemRateCardEntry • OrderItemRateAdjustment	Optional	63.0

Response body for GET

Usage Details

Quote Line Item Usage Details (GET)

Get details of a usage-based product associated with a quote line item.

Here are the details of the information that this API returns:

- Grants and resources for the product, if rates aren't configured.
- Grants, resources, and any configured rates for the product. The rates are returned by the Rate Plan (GET) API.
- Resources that include grants, if applicable, and any negotiated rates for the product in case of a rate override request.

Resource

```
/commerce/quotes/line-items/quoteLineItemId/usage-details
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/commerce/quotes/line-items/0QLXXXXXXXXXXXABC/usage-details
```

Available version

62.0

HTTP methods

GET

Request parameters for GET

Parameter Name	Type	Description	Required or Optional	Available Version
`effectiveDate`	String	Date that's used to search for the applicable rate card entries.	Optional	62.0
`optionalFields`	String[]	Custom fields that you can use to query these objects. • QuoteLineRateCardEntry • QuoteLineRateAdjustment	Optional	62.0

Response body for GET

Usage Details

Response Bodies

Learn more about the available response bodies of Usage Management APIs.

Fields Response

Output representation of the details of the optional fields on the usage-based selling-related objects.

Rate Adjustments

Output representation of the details of a rate adjustment.

Rate Card Entry

Output representation of the details of a rate card entry.

Usage Details

Output representation of the usage details of a quote, an order, or an asset.

Usage Details Error Response

Output representation of the details of an error related to usage details.

Fields Response

Output representation of the details of the optional fields on the usage-based selling-related objects.

JSON Example

```
"fields": {
  "MyCustomDate__c": {
    "displayValue": "2024-09-24",
    "value": "2024-09-24T17:46:30.662Z"
  },
  "MyCustomNumber__c": {
    "displayValue": "20.0",
    "value": 20
  }
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
displayValue	String	Display value of a field.	Big, 63.0	63.0
value	Object	Value of a field in its original data form.	Big, 63.0	63.0

Rate Adjustments

Output representation of the details of a rate adjustment.

JSON Example

```
    "rateAdjustments": [
      {
         "fields": {},
         "lowerBound": 0.0,
         "name": null,
         "negotiatedRateAdjustmentId": null,
         "rateAdjustmentId": "1ENxx0000004C9BGAU",
         "rateAdjustmentType": "Percentage",
         "rateAdjustmentValue": 10.0,
         "tierUnitOfMeasure": "USD",
         "upperBound": 50.0
      }
  ]
```

If the negotiable property value that's associated with a rate card entry is blank, then the data is derived from Product Catalog Management. If it isn't blank, then the data is derived from Rate Management.

Property Name	Type	Description	Filter Group and Version	Available Version
fields	Map<String, Fields Response>	List of optional fields and their values that belong to the rate adjustment object.	Big, 63.0	63.0
lowerBound	Double	Minimum quantity for the adjustment to be applicable.	Small, 63.0	63.0
name	String	Name of the tier.	Small, 63.0	63.0
negotiatedRateAdjustmentId	String	ID of the negotiated rate adjustment.	Small, 63.0	63.0
rateAdjustmentId	String	ID of the rate adjustment.	Small, 63.0	63.0
rateAdjustmentType	String	Type of the rate adjustment.	Small, 63.0	63.0
rateAdjustmentValue	Double	Value of the rate adjustment.	Small, 63.0	63.0
tierUnitOfMeasure	String	Unit of measure representing the tier.	Small, 63.0	63.0

Property Name	Type	Description	Filter Group and Version	Available Version
upperBound	Double	Maximum quantity for the adjustment to be applicable.	Small, 63.0	63.0

Rate Card Entry

Output representation of the details of a rate card entry.

JSON Example

```
{
    "records": [
        {
            "bindingInstanceTargetType": "Product",
            "bindingInstanceType": "Target",
            "chargeForOverages": "Yes",
            "fields": {},
            "name": "Paddle Board",
            "negotiable": "Negotiable,Non-Negotiable",
            "negotiatedRate": 20.0,
            "negotiatedRateCardEntryId": "1ELxx0000004C9JGAU,1ELxx0000004C9KGAU",
            "quantity": 15.0,
            "rate": 5.0,
            "rateAdjustments": [
                {
                    "fields": {},
                    "lowerBound": 0.0,
                    "name": "Tier 1",
                    "negotiatedRateAdjustmentId": "1ENxx0000004C9BGAU",
                    "rateAdjustmentId": "1ENxx0000004C9BGAU",
                    "rateAdjustmentType": "Percentage",
                    "rateAdjustmentValue": 10.0,
                    "tierUnitOfMeasure": "USD",
                    "upperBound": 50.0
                }
            ],
            "rateCardEntryId": "1CJxx0000004C9IGAU,1CJxx0000004C9JGAU",
            "unitOfMeasure": "USD"
        }
    ]
}
```

If the `negotiable` property value is blank, then the data is derived from Product Catalog Management. If it isn't blank, then the data is derived from Rate Management.

Property Name	Type	Description	Filter Group and Version	Available Version
bindingInstanceTargetType	String	Type of the target object that's associated with this transaction.	Big, 63.0	63.0

Property Name	Type	Description	Filter Group and Version	Available Version
bindingInstanceType	String	Type of the binding instance. Valid values are: • Self • Target	Big, 63.0	63.0
chargeForOverages	String	Specifies whether overage is permitted. Valid values are: • Yes • No • NA	Big, 63.0	63.0
fields	Map<String, Fields Response>	List of optional fields and their values that's associated with the rate card entry object.	Big, 63.0	63.0
name	String	Name of the resource.	Big, 63.0	63.0
negotiable	String	Type of the base rate and the tier rate, if applicable. Valid values are: • Negotiable • Non-Negotiable	Big, 63.0	63.0
negotiatedRate	Double	User-overridden overage rate.	Big, 63.0	63.0
negotiatedRateCardEntryId	String	ID of the negotiated rate card entry and the tier rate card entry, if applicable.	Big, 63.0	63.0
quantity	Double	Amount granted for the resource.	Big, 63.0	63.0
rate	Double	Base overage rate.	Big, 63.0	63.0
rateAdjustments	Rate Adjustments[]	List of tiers associated with the rate card entry, if applicable.	Big, 63.0	63.0
rateCardEntryId	String	ID of the base rate card entry and the tier rate card entry, if applicable.	Big, 63.0	63.0
unitOfMeasure	String	Unit of measure of the grant. For example, Unit.	Big, 63.0	63.0

Usage Details

Output representation of the usage details of a quote, an order, or an asset.

JSON example

This sample response shows resources that include grants, if applicable, and resources without rates when you retrieve the usage details of an order item.

```
{
    "records": [
        {
            "bindingInstanceTargetType": null,
            "bindingInstanceType": null,
            "chargeForOverages": "Yes",
            "fields": {},
            "name": "API Calls",
            "negotiable": ",",
            "negotiatedRate": null,
            "negotiatedRateCardEntryId": ",",
            "quantity": 1000.0,
            "rate": null,
            "rateAdjustments": [],
            "rateCardEntryId": ",",
            "unitOfMeasure": null
        }
    ]
}
```

This example shows a sample response without negotiated rates when you retrieve the usage details of an order item.

```
{
    "records": [
        {
            "bindingInstanceTargetType": "Product",
            "bindingInstanceType": "Target",
            "chargeForOverages": "Yes",
            "fields": {}
            "name": "Paddle Board",
            "negotiable": "Negotiable,Non-Negotiable",
            "negotiatedRate": null,
            "negotiatedRateCardEntryId": ",",
            "quantity": 15.0,
            "rate": 5.0,
            "rateAdjustments": [
                {
                    "fields": {},
                    "lowerBound": 0.0,
                    "name": null,
                    "negotiatedRateAdjustmentId": null,
                    "rateAdjustmentId": "1ENxx0000004C9BGAU",
                    "rateAdjustmentType": "Percentage",
                    "rateAdjustmentValue": 10.0,
                    "tierUnitOfMeasure": "USD",
                    "upperBound": 50.0
                }
            ],
            "rateCardEntryId": "1CJxx0000004C9IGAU,1CJxx0000004C9JGAU",
            "unitOfMeasure": "USD"
        }
```

```
    ]
}
```

This sample response shows negotiated rates when you retrieve the usage details of an order item. The negotiated rates are derived from these objects for assets, order items, or quote line items.

- AssetRateCardEntry
- AssetRateAdjustment
- OrderItemRateCardEntry
- OrderItemRateAdjustment
- QuoteLineRateCardEntry
- QuoteLineRateAdjustment

```
{
    "records": [
        {
            "bindingInstanceTargetType": "Product",
            "bindingInstanceType": "Target",
            "chargeForOverages": "Yes",
            "fields": {},
            "name": "Paddle Board",
            "negotiable": "Negotiable,Non-Negotiable",
            "negotiatedRate": 20.0,
            "negotiatedRateCardEntryId": "1ELxx0000004C9JGAU,1ELxx0000004C9KGAU",
            "quantity": 15.0,
            "rate": 5.0,
            "rateAdjustments": [
                {
                    "fields": {},
                    "lowerBound": 0.0,
                    "name": "Tier 1",
                    "negotiatedRateAdjustmentId": "1ENxx0000004C9BGAU",
                    "rateAdjustmentId": "1ENxx0000004C9BGAU",
                    "rateAdjustmentType": "Percentage",
                    "rateAdjustmentValue": 10.0,
                    "tierUnitOfMeasure": "USD",
                    "upperBound": 50.0
                }
            ],
            "rateCardEntryId": "1CJxx0000004C9IGAU,1CJxx0000004C9JGAU",
            "unitOfMeasure": "USD"
        }
    ]
}
```

This sample response shows details of the custom fields when you retrieve the usage details of an order item.

```
{
    "records": [
        {
            "fields": {
                "MyCustomDate__c": {
                    "displayValue": "2024-09-24",
                    "value": "2024-09-24T17:46:30.662Z"
```

```
                },
                "MyCustomNumber__c": {
                    "displayValue": "20.0",
                    "value": 20
                }
            },
            "bindingInstanceTargetType": "Product",
            "bindingInstanceType": "Target",
            "chargeForOverages": "Yes",
            "name": "Massage Therapy",
            "negotiable": "Negotiable,Non-Negotiable",
            "negotiatedRate": 20.0,
            "negotiatedRateCardEntryId": "1ELxx0000004C9JGAU,1ELxx0000004C9KGAU",
            "quantity": 15.0,
            "rate": 5.0,
            "rateAdjustments": [
                {
                    "fields": {
                        "MyCustomString__c": {
                            "displayValue": "My Custom String",
                            "value": "MyCustomString"
                        }
                    },
                    "lowerBound": 0.0,
                    "name": "Tier 1",
                    "negotiatedRateAdjustmentId": "1ENxx0000004C9BGAU",
                    "rateAdjustmentId": "1ENxx0000004C9BGAU",
                    "rateAdjustmentType": "Percentage",
                    "rateAdjustmentValue": 10.0,
                    "tierUnitOfMeasure": "USD",
                    "upperBound": 50.0
                }
            ],
            "rateCardEntryId": "1CJxx0000004C9IGAU,1CJxx0000004C9JGAU",
            "unitOfMeasure": "USD"
        }
    ]
}
```


Property Name	Type	Description	Filter Group and Version	Available Version
errors	Usage Detail Error Response[]	List of errors.	Small, 63.0	63.0
records	Rate Card Entry[]	List of rate card entry records.	Small, 63.0	63.0

Usage Details Error Response

Output representation of the details of an error related to usage details.

JSON Example

```
{
   "errors": [
        {
            "referenceId": "MyOrderItem",
            "errorCode": "INVALID_API_INPUT",
            "message": "Something has failed"
        }
   ]
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`errorCode`	String	Code associated with the error.	Small, 63.0	63.0
`message`	String	Message associated with the error.	Small, 63.0	63.0
`referenceId`	String	Unique ID that's associated with the specific error for tracking and reference purposes.	Small, 63.0	63.0

Usage Management Metadata API Types

Metadata API enables you to access some types and feature settings that you can customize in the user interface.

Flow for Usage Management

Represents the metadata associated with a flow. With Flow, you can create an application that navigates users through a series of screens to query and update records in the database. You can also execute logic and provide branching capability based on user input to build dynamic applications.

SEE ALSO:

Metadata API Developer Guide: Understanding Metadata API

Flow for Usage Management

Represents the metadata associated with a flow. With Flow, you can create an application that navigates users through a series of screens to query and update records in the database. You can also execute logic and provide branching capability based on user input to build dynamic applications.

FlowActionCall

Usage Management exposes additional actionType values for the FlowActionCall Metadata type.

Field Name	Field Type	Description
`actionType`	InvocableActionType (enumeration of type string)	Required. The action type. Additional valid values only for Usage Management include: • `invokeSummaryCreationService`—Invoke the service that creates various summaries, such as usage, ratable, and liable summaries where the usage amount is zero. The service also checks and updates the billing period of the usage entitlement account if the billing period is expired. • `processConsumptionOverages`—Process consumption overages for the usage summary records with `SummaryComplete` status. This action uses the entitlement service to process the overages. • `refreshUsageEntitlementBucket`—Refresh entitlements by evaluating the usage entitlement bucket records and creating a new usage entitlement entry.

CHAPTER 7 Rate Management

In this chapter ...

- Rate Management Standard Objects
- Rate Management Metadata API Types
- Rate Management Business APIs
- Rate Management Standard Invocable Actions

Quote and price products based on predefined rates for future use of the product or service.

SEE ALSO:

Salesforce Help: Rate Management Permission Set Licenses

EDITIONS

Available in: Lightning Experience

Available in: **Enterprise**, **Unlimited**, and **Developer** Editions of Revenue Cloud where Rate Management is enabled

Rate Management Standard Objects

The Rate Management data model provides objects and fields to manage rates and discounts for a product's resource consumption.

PriceBookRateCard

Represents a junction between price book and rate card objects. This object is available in API version 62.0 and later.

RateAdjustmentByAttribute

Represents the adjustments that determine the rate of a resource based on its rate-impacting attributes. These attributes are linked to the usage product record. Rates are then influenced by conditions specified in the Attribute Based Adjustment Condition object. Finally, the charge rate is determined by using the Attribute Based Adjustment Rule object. This object is available in API version 62.0 and later.

RateAdjustmentByTier

Represents the adjustments for the rate of a resource that's determined based on the specified tiers. This object stores information about the type of adjustment used, the value of the adjustment type, and any applicable boundaries. This object is available in API version 62.0 and later.

RateCard

Represents the rules used to rate the consumption of a group of resources within a product. Usage of a resource is billed at a specified rate if the user consumes more than their allowance for a time period. This object is available in API version 62.0 and later.

RateCardEntry

Represents a rule that determines the charge rate for using a product's resource. Each entry is linked to one rate card exclusively, and its activation or deactivation can be controlled by assigning effective dates. This object is available in API version 62.0 and later.

RatingFrequencyPolicy

Represents the policy that defines the frequency at which rating is triggered for the ratable summary records. This object is available in API version 62.0 and later.

RatingRequest

Represents the common run-time parameters, such as context definition and rating procedure for a set of records in the rateable summary table. This object is available in API version 62.0 and later.

RatingRequestBatchJob

Represents a junction between the rating request and batch job objects. This object is available in API version 62.0 and later.

SEE ALSO:

Object Reference for the Salesforce Platform: Overview of Salesforce Objects and Fields

SOAP API Developer Guide: Introduction to SOAP API

PriceBookRateCard

Represents a junction between price book and rate card objects. This object is available in API version 62.0 and later.

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.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Timestamp for when the current user last referred to this record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Timestamp for when the current user last viewed a record related to this record.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Auto-generated identifier for the price book rate card record.
`PriceBookId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Price book ID that's associated with the rate cards IDs. For Quote, Order, and Contracts, the price book IDs identify the associated rate cards. This field is a relationship field. **Relationship Name** PriceBook **Relationship Type** Master-detail **Refers To** Pricebook2 (the master object)

Field	Details
`RateCardId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Rate card ID that's associated with the price book. This field is a relationship field. **Relationship Name** RateCard **Relationship Type** Master-detail **Refers To** RateCard (the detail object)
`RateCardType`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Type of rate card associated with the price book. Valid values are: • `Attribute` • `Base` • `Tier`

Associated Objects

This object has these associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

PriceBookRateCardFeed
Feed tracking is available for the object.

PriceBookRateCardHistory
History is available for tracked fields of the object.

RateAdjustmentByAttribute

Represents the adjustments that determine the rate of a resource based on its rate-impacting attributes. These attributes are linked to the usage product record. Rates are then influenced by conditions specified in the Attribute Based Adjustment Condition object. Finally, the charge rate is determined by using the Attribute Based Adjustment Rule object. This object is available in API version 62.0 and later.

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.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`AdjustmentType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Type of rate adjustment. Valid values are: • `Amount` • `Override` • `Percentage`
`AdjustmentValue`	**Type** double **Properties** Create, Filter, Sort, Update **Description** Value of the rate adjustment based on the selected adjustment type.
`AttributeBasedAdjRuleId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** ID of the attribute based adjustment rule associated with this rate adjustment by attribute record. This field is a relationship field. **Relationship Name** AttributeBasedAdjRule **Refers To** AttributeBasedAdjRule

Field	Details
EffectiveFrom	**Type** dateTime **Properties** Filter, Sort **Description** Date and time when the associated rate card entry comes into effect.
EffectiveTo	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Date and time until when the associated rate card entry remains effective.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Timestamp for when the current user last referred to this record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Timestamp for when the current user last viewed a record related to this record.
Name	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Auto-generated identifier for the rate adjustment by attribute record.
ProductId	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** ID of the product whose resource is being used as the associated rate card entry.

Field	Details
	This field is a relationship field. **Relationship Name** Product **Refers To** Product2
`ProductSellingModelId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** ID of the product selling model for the associated rate card entry. This field is a relationship field. **Relationship Name** ProductSellingModel **Refers To** ProductSellingModel
`RateCardEntryId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** ID of the rate card entry associated with this rate adjustment by attribute record. This field is a relationship field. **Relationship Name** RateCardEntry **Relationship Type** Master-detail **Refers To** RateCardEntry (the master object)
`RateCardEntryStatus`	**Type** picklist **Properties** Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort **Description** Status of the rate card entry associated with this rate adjustment by attribute. Valid values are: • `Active`

Field	Details
	• `Draft` • `Inactive` The default value is `Draft`. Available in API version 63.0 and later.
`RateCardId`	**Type** reference **Properties** Filter, Group, Sort **Description** ID of the rate card of the associated rate card entry. This field is a relationship field. **Relationship Name** RateCard **Refers To** RateCard
`RateUnitOfMeasureId`	**Type** reference **Properties** Filter, Group, Sort **Description** ID of the standard unit of measure record of the associated rate card entry. This field is a relationship field. **Relationship Name** RateUnitOfMeasure **Refers To** UnitOfMeasure
`RateUnitOfMeasureName`	**Type** string **Properties** Filter, Group, Sort **Description** Name of the standard unit of measure record of the associated rate card entry.
`UsageResourceId`	**Type** reference **Properties** Filter, Group, Sort **Description** ID of the resource selected for the associated rate card entry.

Field	Details
	This field is a relationship field. **Relationship Name** UsageResource **Refers To** UsageResource

Associated Objects

This object has these associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

RateAdjustmentByAttributeFeed
Feed tracking is available for the object.

RateAdjustmentByAttributeHistory
History is available for tracked fields of the object.

RateAdjustmentByTier

Represents the adjustments for the rate of a resource that's determined based on the specified tiers. This object stores information about the type of adjustment used, the value of the adjustment type, and any applicable boundaries. This object is available in API version 62.0 and later.

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.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`AdjustmentType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Type of rate adjustment. Valid values are: • `Amount` • `Override`

Field	Details
	• `Percentage`
`AdjustmentValue`	**Type** double **Properties** Create, Filter, Sort, Update **Description** Value of the rate adjustment based on the selected adjustment type.
`EffectiveFrom`	**Type** dateTime **Properties** Filter, Sort **Description** Date and time when the associated rate card entry comes into effect.
`EffectiveTo`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Date and time until when the associated rate card entry remains effective.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Timestamp for when the current user last referred to this record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Timestamp for when the current user last viewed a record related to this record.
`LowerBound`	**Type** double **Properties** Create, Filter, Sort, Update

Field	Details
	Description Minimum quantity of the rate adjustment value that can be applied to the record. This value must be a positive integer and must be less than the upper bound of the tier.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Auto-generated identifier for the rate adjustment by tier record.
`ProductId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** ID of the product whose resource is being used as the associated rate card entry. This field is a relationship field. **Relationship Name** Product **Refers To** Product2
`ProductSellingModelId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** ID of the product selling model for the associated rate card entry record. This field is a relationship field. **Relationship Name** ProductSellingModel **Refers To** ProductSellingModel
`RateCardEntryId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** ID of the rate card entry associated with this rate adjustment by tier record.

Field	Details
	This field is a relationship field. **Relationship Name** RateCardEntry **Relationship Type** Master-detail **Refers To** RateCardEntry (the master object)
`RateCardEntryStatus`	**Type** picklist **Properties** Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort **Description** Status of the rate card entry associated with this rate adjustment by tier. Valid values are: • `Active` • `Draft` • `Inactive` The default value is `Draft`. Available in API version 63.0 and later.
`RateCardId`	**Type** reference **Properties** Filter, Group, Sort **Description** ID of the rate card of the associated rate card entry. This field is a relationship field. **Relationship Name** RateCard **Refers To** RateCard
`RateUnitOfMeasureId`	**Type** reference **Properties** Filter, Group, Sort **Description** ID of the standard unit of measure record of the associated rate card entry. This field is a relationship field.

Field	Details
	Relationship Name RateUnitOfMeasure **Refers To** UnitOfMeasure
`RateUnitOfMeasureName`	**Type** string **Properties** Filter, Group, Sort **Description** Name of the standard unit of measure record of the associated rate card entry.
`UpperBound`	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** Maximum quantity of the rate adjustment value that can be applied to the record. This value must be a positive double and must be greater than the lower bound of the tier.
`UsageResourceId`	**Type** reference **Properties** Filter, Group, Sort **Description** ID of the resource selected for the associated rate card entry. This field is a relationship field. **Relationship Name** UsageResource **Refers To** UsageResource

Associated Objects

This object has these associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

RateAdjustmentByTierFeed
Feed tracking is available for the object.

RateAdjustmentByTierHistory
History is available for tracked fields of the object.

RateCard

Represents the rules used to rate the consumption of a group of resources within a product. Usage of a resource is billed at a specified rate if the user consumes more than their allowance for a time period. This object is available in API version 62.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Description`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Description about the rate card.
`EffectiveFrom`	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** Date and time when the rate card becomes effective.
`EffectiveTo`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** Date and time until when the rate card remains effective.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Timestamp for when the current user last referred to this record.
`LastViewedDate`	**Type** dateTime

Field	Details
	Properties Filter, Nillable, Sort **Description** Timestamp for when the current user last viewed a record related to this record.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** Name of the rate card.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** ID of the user who created the record. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
`Type`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Type of rate card. Valid values are: • `Attribute` • `Base` • `Tier`

Associated Objects

This object has these associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

RateCardFeed
Feed tracking is available for the object.

RateCardHistory
History is available for tracked fields of the object.

RateCardShare
Sharing is available for the object.

RateCardEntry

Represents a rule that determines the charge rate for using a product's resource. Each entry is linked to one rate card exclusively, and its activation or deactivation can be controlled by assigning effective dates. This object is available in API version 62.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`DefaultUnitOfMeasureClassId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** ID of the default unit of measure classification record associated with this rate card entry. This field is a relationship field. **Relationship Name** DefaultUnitOfMeasureClass **Refers To** UnitOfMeasureClass
`DefaultUnitOfMeasureId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** ID of the default unit of measure record associated with this rate card entry. This field is a relationship field. **Relationship Name** DefaultUnitOfMeasure

Field	Details
	Refers To UnitOfMeasure
EffectiveFrom	**Type** dateTime **Properties** Create, Filter, Sort, Update **Description** Date and time when the rate card entry comes into effect.
EffectiveTo	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** Date and time until when the rate card entry remains effective.
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Timestamp for when the current user last referred to this record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Timestamp for when the current user last viewed a record related to this record.
Name	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Auto-generated identifier for the rate card entry record.
ProductId	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description ID of the product whose resource is being used as a rate card entry. This field is a relationship field. **Relationship Name** Product **Refers To** Product2
`ProductSellingModelId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** ID of the product selling model associated with this rate card entry. This field is a relationship field. **Relationship Name** ProductSellingModel **Refers To** ProductSellingModel
`Rate`	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** Value of the rate card entry.
`RateCardId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** ID of the rate card associated with this rate card entry. This field is a relationship field. **Relationship Name** RateCard **Relationship Type** Master-detail **Refers To** RateCard (the master object)

Field	Details
`RateCardType`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** Type of rate card associated with this rate card entry. Valid values are: • `Attribute` • `Base` • `Tier` Available in API version 63.0 and later.
`RateNegotiation`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update **Description** Type of rate negotiation applicable to the rate card entry. Valid values are: • `Negotiable` • `NonNegotiable` The default value is `Negotiable`. Available in API version 63.0 and later.
`RateUnitOfMeasureClassId`	**Type** reference **Properties** Filter, Group, Sort **Description** ID of the unit of measure classification record associated with this rate card entry. This field is a relationship field. **Relationship Name** RateUnitOfMeasureClass **Refers To** UnitOfMeasureClass
`RateUnitOfMeasureId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update

Field	Details
	Description ID of the standard unit of measure record associated with this rate card entry. This field is a relationship field. **Relationship Name** RateUnitOfMeasure **Refers To** UnitOfMeasure
`RateUnitOfMeasureName`	**Type** string **Properties** Filter, Group, Sort **Description** Name of the standard unit of measure record of the associated rate card entry.
`Status`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update **Description** Status of the rate card entry. • `Active` • `Draft` • `Inactive` The default value is `Draft`. Available in API version 63.0 and later.
`UsageProductId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** ID of the product associated with the resource for which the rate is specified. This field is a relationship field. **Relationship Name** UsageProduct **Refers To** Product2
`UsageResourceId`	**Type** reference

Field	Details
	Properties Create, Filter, Group, Sort, Update **Description** ID of the resource associated with this rate card entry. This field is a relationship field. **Relationship Name** UsageResource **Refers To** UsageResource

Associated Objects

This object has these associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

RateCardEntryFeed
Feed tracking is available for the object.

RateCardEntryHistory
History is available for tracked fields of the object.

RatingFrequencyPolicy

Represents the policy that defines the frequency at which rating is triggered for the ratable summary records. This object is available in API version 62.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Timestamp for when the current user last referred to this record.
`LastViewedDate`	**Type** dateTime

Field	Details
	Properties Filter, Nillable, Sort **Description** Timestamp for when the current user last viewed a record related to this record.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Auto-generated identifier for the rating frequency policy record.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** ID of the user who created the record. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
`ProductId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** ID of the product for which the rating policy is defined. This field is a relationship field. **Relationship Name** Product **Refers To** Product2
`RatingDelayDuration`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description Duration of delay—in hours—post the billing period after which the rating is to be triggered.
`RatingPeriod`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Period for which the usage of a product and usage resource combination are to be rated. Valid values are: • `Daily` • `Monthly`
`UsageResourceId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** ID of the usage resource for which the rating policy is defined. This field is a relationship field. **Relationship Name** UsageResource **Refers To** UsageResource

Associated Objects

This object has these associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

RatingFrequencyPolicyFeed
Feed tracking is available for the object.

RatingFrequencyPolicyHistory
History is available for tracked fields of the object.

RatingFrequencyPolicyShare
Sharing is available for the object.

RatingRequest

Represents the common run-time parameters, such as context definition and rating procedure for a set of records in the rateable summary table. This object is available in API version 62.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`ContextDefinition`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** Context definition that's used for context instance creation, which encapsulates all aggregated records that are stamped for the rating request.
`ContextMapping`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Context mapping that's used for context instance creation, which encapsulates all aggregated records that are stamped for the rating request. If no ID is provided, default context mapping is used.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Timestamp for when the current user last referred to this record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Timestamp for when the current user last viewed a record related to this record.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort

Field	Details
	Description Auto-generated identifier for the rating request record.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** ID of the user who created the record. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
`RatingProcedureName`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** Procedure name that's used to rate the aggregated records that are stamped for rating request.
`Status`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Status of the rating request. Valid values are: • `Failed` • `Pending` • `RatingComplete` • `RatingInProgress` • `ReadyForRating`

Associated Objects

This object has these associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

RatingRequestFeed
Feed tracking is available for the object.

RatingRequestHistory
History is available for tracked fields of the object.

RatingRequestShare
Sharing is available for the object.

RatingRequestBatchJob

Represents a junction between the rating request and batch job objects. This object is available in API version 62.0 and later.

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.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`BatchJobId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** ID of the batch job that triggered the rating request on the aggregated records. This field is a relationship field. **Relationship Name** BatchJob **Refers To** BatchJob
`ErrorCode`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Error code that defines the batch job failure. Valid values are: • `BadRequest` • `InternalError`

Field	Details
`ErrorMessage`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Error message that describes the cause of the batch job failure.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Timestamp for when the current user last referred to this record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Timestamp for when the current user last viewed a record related to this record.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Name of the rating request batch job record.
`RatingRequestId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** ID of the rating request record associated with the batch job. This field is a relationship field. **Relationship Name** RatingRequest **Relationship Type** Master-detail **Refers To** RatingRequest (the master object)

Associated Objects

This object has these associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

RatingRequestBatchJobFeed
Feed tracking is available for the object.

RatingRequestBatchJobHistory
History is available for tracked fields of the object.

Rate Management Metadata API Types

Metadata API enables you to access some types and feature settings that you can customize in the user interface.

IndustriesRatingSettings
Represents the settings for Rate Management.

Flow for Rate Management
Represents the metadata associated with a flow. With Flow, you can create an application that takes users through a series of pages to query and update the records in the database. You can also run logic and provide branching capability based on user input to build dynamic applications.

IndustriesRatingSettings

Represents the settings for Rate Management.

Parent Type and Manifest Access

This type extends the Metadata metadata type and inherits its `fullName` field.

In the package manifest, all the settings metadata types for the org are accessed using the "Settings" name. See Settings for more details.

File Suffix and Directory Location

The `IndustriesRatingSettings` values are stored in the `IndustriesRating.settings` file in the `settings` folder. The `.settings` files are different from other named components, because there's only one settings file for each settings component.

Version

IndustriesRatingSettings components are available in API version 62.0 and later.

Special Access Rules

This metadata type is available with Rate Management.

Fields

Field Name	Description
enableRating	**Field Type** boolean **Description** Indicates whether to enable Rate Management (`true`) or not (`false`). The default value is `false`.
enableRatingWaterfall	**Field Type** boolean **Description** Indicates whether to enable Rating Waterfall (`true`) or not (`false`). The default value is `false`. Rating Waterfall provides insights into the rating data, which you can synchronize with your rating lookup tables.
enableRatingWaterfallPersistence	**Field Type** boolean **Description** Indicates whether to enable Rating Waterfall Persistence (`true`) or not (`false`). The default value is `false`. Rating Waterfall Persistence stores rating data, which you can use to enhance the internal processes and increase efficiency.

Declarative Metadata Sample Definition

The following is an example of an IndustriesRatingSettings component.

```
<IndustriesRatingSettings xmlns="http://soap.sforce.com/2006/04/metadata">
     <enableRating>true</enableRating>
     <enableRatingWaterfall>true</enableRatingWaterfall>
     <enableRatingWaterfallPersistence>true</enableRatingWaterfallPersistence>
</IndustriesRatingSettings>
```

The following is an example `package.xml` that references the previous definition.

```
<?xml version="1.0" encoding="UTF-8"?>
<Package xmlns="http://soap.sforce.com/2006/04/metadata">
    <types>
        <members>IndustriesRating</members>
        <name>Settings</name>
    </types>
    <version>63.0</version>
</Package>
```

Wildcard Support in the Manifest File

The wildcard character `*` (asterisk) in the `package.xml` manifest file doesn't apply to metadata types for feature settings. The wildcard applies only when retrieving all settings, not for an individual setting. For details, see Settings. For information about using the manifest file, see Deploying and Retrieving Metadata with the Zip File.

Flow for Rate Management

Represents the metadata associated with a flow. With Flow, you can create an application that takes users through a series of pages to query and update the records in the database. You can also run logic and provide branching capability based on user input to build dynamic applications.

FlowActionCall

Rate Management exposes additional actionType values for the FlowActionCall metadata type.

Field Name	Field Type	Description
`actionType`	InvocableActionType (enumeration of type string)	Required. The action type. Additional valid values for Rate Management include: • `invokeRatingService`—Invoke the rating service to rate the usage records.

Rate Management Business APIs

Use the Rate Management Business APIs to get rate plan and persisted rating waterfall details.

Resources
Learn more about the available Rate Management API resources.

Response Bodies
Learn more about the available response bodies of Rate Management APIs.

SEE ALSO:

Connect REST API Developer Guide: Introduction

Resources

Learn more about the available Rate Management API resources.

Rate Plan (GET)
Get a rate plan for a specified set of context input. Use this API to retrieve rate cards, rate card entries, and related adjustments based on the filter criteria for the context input.

Rating Waterfall (GET)

Get the persisted rating waterfall that stores the process logs. Rating waterfall provides insights into the internal rating process.

Rate Plan (GET)

Get a rate plan for a specified set of context input. Use this API to retrieve rate cards, rate card entries, and related adjustments based on the filter criteria for the context input.

Keep these considerations in mind when you use this API.

- This API request supports one pricebook and one sellable product.
- The ID of the product is required to invoke this API.
- Invoke this API even if a hydrated context is available to fetch the rate card, rate card entry, and adjustment details.

Special Access Rules

The org must have the Rate Management: Run Time User permission set to use this API. Additionally, the org must also have a default usage rating procedure defined in the Revenue Settings from Setup.

Resource

```
/connect/core-rating/rate-plan
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/core-rating/rate-plan
?contextId=858a3ad3e5a0e5c319652a6ab92f6fdb2b4fa8be72b390506d014596c6da62c9&procedure
ApiName=SampleProcedure
```

Available version

62.0

HTTP methods

GET

Request parameters for GET

Parameter Name	Type	Description	Required or Optional	Available Version
`contextId`	String	ID of the context to specify as an input to the procedure.	Required	62.0
`procedureApiName`	String	API name of the procedure to be executed.	Required	62.0

Response body for GET

Rate Plan Response

Rating Waterfall (GET)

Get the persisted rating waterfall that stores the process logs. Rating waterfall provides insights into the internal rating process.

Resource

```
/connect/core-pricing/waterfall/lineItemId/executionId
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/core-pricing/waterfall/Gold/2yHbNNEFOZr9jAe4gHS7?tagsToFilter=UnitPrice&usageType=Rating
```

Available version

62.0

HTTP methods

GET

Query parameters

Name	Type	Description	Required or Optional	Available Version
`tagsToFilter`	String	Comma-separated tags to filter.	Optional	62.0
`usageType`	String	Usage type of the waterfall log record. Valid values are: • `Rating` • `Pricing`—Specifies that the record type is `Pricing`. If this value is specified, the API creates a log of pricing waterfall. See Pricing Waterfall. The default value is `Pricing`.	Optional	62.0

Response body for GET

Line Item Waterfall Response

Response Bodies

Learn more about the available response bodies of Rate Management APIs.

Adjustment Details

Output representation of a rate adjustment request.

Line Item Waterfall Response

Output representation of the line item waterfall response.

Rate Plan Response

Output representation of the details of a rate plan.

Rating Error Response

Output representation of the error details related to the API request.

Rating Waterfall Response

Output representation of a rating waterfall request.

Adjustment Details

Output representation of a rate adjustment request.

JSON example

```
 "pricingElement": {
      "adjustments": [{
      "adjustmentType": null,
      "adjustmentValue": null
    }],
      "name": "List Price",
      "elementType": "ListPrice"
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
adjustments	Map<String, Object>[]	Details of the rate element.	Small, 60.0	60.0
description	String	Description of the rate element.	Small, 60.0	60.0
elementType	String	Type of the rate element.	Small, 60.0	60.0
name	String	Name of the rate element.	Small, 60.0	60.0

Line Item Waterfall Response

Output representation of the line item waterfall response.

JSON example

```
{
       "currencyCode": "USD",
       "error": null,
       "executionEndTimestamp": "2023-07-31T20:11:29.625Z",
       "executionId": "gdLVwn2x1uats2xWMAjV",
       "executionStartTimestamp": null,
       "lineItemId": "item1",
       "success": true,
       "usageType":"Rating",
       "output": {
          "quantity": "10",
          "netUnitPrice": "10",
          "subtotal": "100"
        },
        "waterfall": []
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`contextDefinitionVersionId`	String	Context definition version ID of the rating procedure.	Small, 62.0	62.0
`contextMappingId`	String	Context mapping ID of the record.	Small, 62.0	62.0
`currencyCode`	String	Currency code. For example, USD or INR.	Small, 62.0	62.0
`error`	Rating Error Response	Details of any errors.	Small, 62.0	62.0
`executionEndTimestamp`	String	End timestamp of procedure execution.	Small, 62.0	62.0
`executionId`	String	Execution ID of a particular execution of a rating procedure.	Small, 62.0	62.0
`executionStartTimestamp`	String	Start timestamp of procedure execution.	Small, 62.0	62.0
`lineItemId`	String	Line item ID for which the price is being calculated.	Small, 62.0	62.0
`output`	Map<String, Object>	Output of the rating procedure.	Small, 62.0	62.0
`success`	Boolean	Indicates whether the API request is successful (`true`) or not (`false`).	Small, 62.0	62.0
`usageType`	String	Usage type of the waterfall log record.	Small, 62.0	62.0
`waterfall`	Rating Waterfall Response[]	Details of the rating waterfall.	Small, 62.0	62.0

Rate Plan Response

Output representation of the details of a rate plan.

JSON example

```
{
  "success": true,
  "executionId" : "a521d592-71c3-4db3-8048-r64504df1605",
  "error": {}
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`error`	Rating Error Response[]	Error response for the API request, if any.	Small, 62.0	62.0
`executionId`	String	ID of the procedure execution record.	Small, 62.0	62.0
`success`	Boolean	Indicates whether the request is successful (`true`) or not (`false`)	Small, 62.0	62.0

Rating Error Response

Output representation of the error details related to the API request.

Property Name	Type	Description	Filter Group and Version	Available Version
`errorCode`	String	Code indicating the type of error.	Small, 62.0	62.0
`message`	String	Message stating the reason for error, if any.	Small, 62.0	62.0

Rating Waterfall Response

Output representation of a rating waterfall request.

JSON example

```
{
  "inputParameters": {
    "productId": "01txx0000006i2SAAQ",
    "pricebookId": "01sxx0000005ptpAAA",
    "pricingModelType": "OneTime"
  },
  "fieldToTagNameMapping": {
    "Product2Id": "ItemProduct",
    "Subtotal": "Subtotal",
    "Pricebook2Id": "Pricebook",
    "Quantity": "ItemQuantity",
    "LineItemId": "SalesTransactionSource",
    "ListPrice": "ItemListPrice"
  },
  "sequence": 0,
  "outputParameters": {
    "listPrice": "10"
  },
  "pricingElement": {
    "adjustments": [
      {
        "adjustmentType": null,
        "adjustmentValue": null
      }
```

```
    ],
    "name": "List Price"
  }
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
fieldToTagName Mapping	Map<String, String>	Mappings of field to tag names.	Small, 62.0	62.0
input Parameters	Map<String, Object>	Parameters of rating element input.	Small, 62.0	62.0
output Parameters	Map<String, Object>	Parameters of rating element output.	Small, 62.0	62.0
pricing Element	Adjustment Details	Details of the rate adjustment of a rating element.	Small, 62.0	62.0
sequence	Integer	Sequence of rating element execution.	Small, 62.0	62.0

Rate Management Standard Invocable Actions

Learn more about the standard invocable actions available with Rate Management.

Invoke Rating Service Action
Invoke the rating service to rate the usage records.

SEE ALSO:

Actions Developer Guide: Overview
REST API Developer Guide: Invocable Actions Standard
Metadata API Developer Guide: Understanding Metadata API

Invoke Rating Service Action

Invoke the rating service to rate the usage records.

The Invoke Rating Service action acts as a connector between batch management and the rating service. This action is available in API version 62.0 and later.

Special Access Rules

The Invoke Rating Service action is available in Enterprise, Unlimited, and Developer Editions where Rate Management is enabled.

Supported REST HTTP Methods

URI

`/services/data/v63.0/actions/standard/invokeRatingService`

Formats

JSON, XML

HTTP Methods

POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`attributeRateCardID`	**Type** string **Description** ID of the rate card that's used to define adjustments based on the attributes that impact the rate.
`baseRateCardID`	**Type** string **Description** ID of the rate card that includes the base rate for the resource to be rated, based on its consumption.
`contextDefinitionId`	**Type** string **Description** Required. ID of the context definition that's used to create the context instance.
`contextMappingID`	**Type** string **Description** ID of the context mapping that maps a standard object, context definition object, or any other input data source to the node that's defined in the context definition.
`isSkipWaterfall`	**Type** boolean **Description** Indicates whether to skip the generation of price waterfall data (`true`) or not (`false`).

Input	Details
procedureName	**Type** string **Description** Name of the rating procedure that's used to calculate the rates.
recordID	**Type** reference **Description** Required. ID of the usage ratable summary record to be rated.
tierRateCardID	**Type** string **Description** ID of the rate card that's used to define adjustments for different tiers of a resource.

Outputs

None

Example

POST

This example shows a sample request for the Invoke Rating Service action.

```
{
  "inputs": [
    {
      "recordIDs": "56jSB000002Bn12CXC",
      "contextMappingId": "11jSB000002Bn13YAC",
      "contextDefinitionId": "11OSB0000000WSv2AM",
      "procedureName": "Invoke Rate",
      "isSkipWaterfall": false,
      "baseRateCardID": "11jSB000002Bn13YAC",
      "tierRateCardID": "fgjjSB0sdf987dsf",
      "attributeRateCardID": "asdfgh563034lk"
    }
  ]
}
```

This example shows a sample response for the Invoke Rating Service action.

```
[
    {
        "actionName": "invokeRatingService",
        "errors": null,
```

```
        "isSuccess": true,
        "outputValues": {}
    }
]
```

CHAPTER 8 Dynamic Revenue Orchestrator

In this chapter ...

- Dynamic Revenue Orchestrator Standard Objects
- Dynamic Revenue Orchestrator Standard Invocable Actions
- Dynamic Revenue Orchestrator Metadata API Types
- Callouts in Dynamic Revenue Orchestrator

Get visibility into a product's fulfillment journey. Also, get a complete view of the entire fulfillment design that includes processes such as order decomposition, fulfillment plans, and jeopardy management.

EDITIONS

Available in: **Enterprise**, **Unlimited**, and **Developer** Editions

Dynamic Revenue Orchestrator Standard Objects

The Dynamic Revenue Orchestrator data model provides objects and fields to manage details of a product's fulfillment.

AssetFulfillmentDecomp

Represents the relationship between an ordered asset and its corresponding fulfillment asset. This object is available in API version 62.0 and later.

FulfillmentAsset

Represents an instance of a technical product used to provide a customer asset. This object is available in API version 61.0 and later.

FulfillmentAssetAttribute

Represents an attribute of a fulfillment asset. This object is available in API version 61.0 and later.

FulfillmentAssetRelationship

Represents a relationship between two fulfillment assets. This object is available in API version 61.0 and later.

FulfillmentFalloutRule

Represents the fulfillment fallout handling rule. This object is available in API version 61.0 and later.

FulfillmentLineAttribute

Represents an attribute of a fulfillment order line. This object is available in API version 61.0 and later.

FulfillmentLineRel

Represents a relationship between two fulfillment order lines. This object is available in API version 61.0 and later.

FulfillmentLineSourceRel

Represents the relationship between a fulfillment order line and its decomposition source. This object is available in API version 61.0 and later.

FulfillmentOrder

Represents a group of products, fees, and delivery charges on a single order that share the same fulfillment location, delivery method, and recipient. The FulfillmentOrderLineItems belonging to a FulfillmentOrder are associated with OrderItemSummary objects belonging to a single OrderSummary. This object is available in API version 48.0 and later.

FulfillmentOrderLineItem

Represents a product or delivery charge belonging to a FulfillmentOrder. Corresponds to an OrderItemSummary. This object is available in API version 48.0 and later.

FulfillmentPlan

Represents a set of steps to be created to fulfill the order. This object is available in API version 61.0 and later.

FulfillmentStep

Represents a task that's required to perform a certain action as part of order fulfillment. This task can be manual or automated. This object is available in API version 61.0 and later.

FulfillmentStepDefinition

Represents a definition of a step that must be executed during fulfillment orchestration. This object is available in API version 61.0 and later.

FulfillmentStepDefinitionGroup

Represents a set of fulfillment step definitions. This object is available in API version 61.0 and later.

FulfillmentStepDependency

Represents a dependency between tasks by defining the order between a task and one that depends on it. This object is available in API version 61.0 and later.

FulfillmentStepDependencyDef

Represents a dependency that must be created between two fulfillment step records. This object is available in API version 62.0 and later.

FulfillmentStepJeopardyRule

Represents the duration and tolerance for the step in the fulfillment process to allow the overall tracking of rules and risks. This object is available in API version 61.0 and later.

FulfillmentStepSource

Represents a link between a fulfillment step and the corresponding order lines. This object is available in API version 61.0 and later.

FulfillmentTaskAssignmentRule

Represents a set of actions that assign a task to a user or queue. This object is available in API version 63.0 and later.

FulfillmentWorkspace

Represents a visual designer for fulfillment plans that can have multiple step groups and their dependencies. This object is available in API version 61.0 and later.

FulfillmentWorkspaceItem

Represents information about the attributes that are used in the definition for a fulfillment step group. This object is available in API version 61.0 and later.

Order

Represents an order associated with a contract or an account.

OrderAction

Indicates the type of order, such as a new sale or a cancellation. This object is available in API version 55.0 and later.

ProductDecompEnrichmentRule

Represents mappings between fields and attributes. Enrichment rules are part of a decomposition rule, and are used to propagate data to fulfillment order lines. This object is available in API version 61.0 and later.

ProductFulfillmentDecompRule

Represents a rule that determines how an order is broken into sub-orders with specific technical details that help in order fulfillment. It can be applied to a commercial or a technical product. This object is available in API version 61.0 and later.

ProductFulfillmentScenario

Represents a link between a product and the corresponding group of fulfillment steps that's necessary to fulfill that product. This object is available in API version 61.0 and later.

SalesTransactionFulfillReq

Represents the statuses of all the sub-orders that belong to the selected commercial or technical product. This object is available in API version 62.0 and later.

ValTfrm

Represents mappings between fields and attributes. Enrichment rules are part of a decomposition rule, and are used to propagate data to fulfillment order lines. This object is available in API version 61.0 and later.

ValTfrmGrp

Represents a rule that determines how an order is broken into sub-orders with specific technical details that help in order fulfillment. The rule can be applied to a commercial or a technical product. This object is available in API version 61.0 and later.

SEE ALSO:

Object Reference for the Salesforce Platform: Overview of Salesforce Objects and Fields

SOAP API Developer Guide: Introduction to SOAP API

AssetFulfillmentDecomp

Represents the relationship between an ordered asset and its corresponding fulfillment asset. This object is available in API version 62.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`FulfillmentSourceAssetId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The identifier of the relationship between an asset and a fulfillment asset. This field is a polymorphic relationship field. **Relationship Name** FulfillmentSourceAsset **Refers To** Asset, FulfillmentAsset
`FulfillmentTargetAssetId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The identifier of the target asset that's being fulfilled. This field is a relationship field. **Relationship Name** FulfillmentTargetAsset **Refers To** FulfillmentAsset
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name of the asset that's being fulfilled.

Field	Details
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The ID of the user who created the request record. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
`RelationshipType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The type of relationship between an asset and a fulfillment asset. Valid values are: • `SourceBundleRoot` • `SourceLineItem`
`SegmentIdentifier`	**Type** string **Properties** Create, Group, Nillable, Sort, **Description** The ID of the ramp segment associated with the asset state period. This field is available in API version 63.0 and later in orgs that have Revenue Cloud when the Ramp Deals setting is enabled.
`Status`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The status of the fulfillment asset. Valid values are: • `Active` • `Inactive`

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

AssetFulfillmentDecompShare on page 1684
Sharing is available for the object.

FulfillmentAsset

Represents an instance of a technical product used to provide a customer asset. This object is available in API version 61.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`AccountId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The identifier of the account. This field is a relationship field. **Relationship Name** Account **Relationship Type** Lookup **Refers To** Account
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **For internal use only.**
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort

Field	Details
	For internal use only.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** The name of the fulfillment asset.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** For internal use only. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User
`ProductId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The identifier of the corresponding product. This field is a relationship field. **Relationship Name** Product **Relationship Type** Lookup **Refers To** Product2
`Quantity`	**Type** double **Properties** Create, Filter, Nillable, Sort, Update

Field	Details
	Description For internal use only.
`Status`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The status of the fulfillment asset. Valid values are: • `Active` • `InActive`
`UnitOfMeasureId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort **Description** The unit of measure for the asset quantity such as, unit, gallon, ton, or case. This field is available in API version 63.0 and later. This field is a relationship field. **Relationship Name** UnitOfMeasure **Refers To** UnitOfMeasure

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

FulfillmentAssetShare on page 1684
Sharing is available for the object.

FulfillmentAssetAttribute

Represents an attribute of a fulfillment asset. This object is available in API version 61.0 and later.

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.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `query()`, `update()`, `upsert()`

Fields

Field	Details
`AttributeDefinitionId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The attribute definition in the catalog. This field is a relationship field. **Relationship Name** AttributeDefinition **Relationship Type** Lookup **Refers To** AttributeDefinition
`AttributeName`	**Type** string **Properties** Filter, Group, idLookup, Nillable, Sort **Description** The name of the attribute.
`AttributePicklistValueId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** For internal use only. This field is a relationship field. **Relationship Name** AttributePicklistValue **Relationship Type** Lookup **Refers To** AttributePicklistValue

Field	Details
`AttributeValue`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The value of the attribute.
`ExternalId`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** For internal use only.
`FulfillmentAssetId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The identifier of the fulfillment asset. This field is a relationship field. **Relationship Name** FulfillmentAsset **Relationship Type** Master-detail **Refers To** FulfillmentAsset (the master object)

FulfillmentAssetRelationship

Represents a relationship between two fulfillment assets. This object is available in API version 61.0 and later.

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.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`AssociatedFulfillAssetRole`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The role of the associated fulfillment asset. Valid values are: • `BundleComponent` • `ClassificationComponent`—Product Classification Component
`AssociatedFulfillmentAssetId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The name of the associated fulfillment asset. This field is a relationship field. **Relationship Name** AssociatedFulfillmentAsset **Refers To** FulfillmentAsset
`MainFulfillmentAssetId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The name of the primary fulfillment asset. This field is a relationship field. **Relationship Name** MainFulfillmentAsset **Relationship Type** Master-detail **Refers To** FulfillmentAsset (the master object)
`MainFulfillmentAssetRole`	**Type** picklist

Field	Details
	Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The role of the primary fulfillment asset. Valid value is `Bundle Parent`.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** The name of the fulfillment asset relationship.
`ProductRelationshipTypeId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The type of relationship between two assets. This field is a relationship field. **Relationship Name** ProductRelationshipType **Refers To** ProductRelationshipType

FulfillmentFalloutRule

Represents the fulfillment fallout handling rule. This object is available in API version 61.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`ErrorCode`	**Type** string

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update **Description** The failure error code of the fulfillment step that's associated with the rule.
`FalloutQueueId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The fallout queue that's associated with the fallout task. This field is available in API version 62.0 and later. This field is a relationship field. **Relationship Name** FalloutQueue **Refers To** Group
`FlowDefinitionName`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The name of the flow definition that's associated with the `AutoTask` type of fulfillment step.
`IntegrationDefinitionId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The integration definition that's associated with the `Callout` type of fulfillment step. This field is a relationship field. **Relationship Name** IntegrationDefinition **Relationship Type** Lookup **Refers To** IntegrationProviderDef

Field	Details
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name of the flow definition that's associated with the `AutoTask` type of fulfillment step.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The ID of the user who created the fulfillment record. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User
`RetriesAllowed`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The maximum number of times a retry policy is run before the fulfillment step is considered failed.
`RetryIntervals`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The interval after which the selected retry policy is run when the fulfillment step fails. This field is available in API version 62.0 and later.
`RetryPolicy`	**Type** picklist

Field	Details
	Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Specifies the retry policy used when the fulfillment step fails. Valid value is: • `Immediate` • `Monotonous` • `Staggered`
`StepType`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Specifies the type of fulfillment step associated with the fallout rule. Valid values are: • `AutoTask` • `Callout` • `ManualTask` • `Milestone` • `Pause`

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

FulfillmentFalloutRuleHistory on page 1680
History is available for tracked fields of the object.

FulfillmentFalloutRuleShare on page 1684
Sharing is available for the object.

FulfillmentLineAttribute

Represents an attribute of a fulfillment order line. This object is available in API version 61.0 and later.

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.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `query()`, `update()`, `upsert()`

Fields

Field	Details
AttributeDefinitionId	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** A unique identifier for the attribute definition in the catalog. This field is a relationship field. **Relationship Name** AttributeDefinition **Refers To** AttributeDefinition
AttributeName	**Type** string **Properties** Filter, Group, idLookup, Nillable, Sort **Description** The name of the attribute.
AttributePicklistValueId	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** For internal use only. This field is a relationship field. **Relationship Name** AttributePicklistValue **Refers To** AttributePicklistValue
AttributeValue	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The value of the attribute.

Field	Details
`ExternalId`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ID that uniquely identifies the relationship in an external data source.
`FulfillmentOrderLineItemId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** A unique identifier for the fulfillment order line item. This field is a relationship field. **Relationship Name** FulfillmentOrderLineItem **Relationship Type** Master-detail **Refers To** FulfillmentOrderLineItem (the master object)

FulfillmentLineRel

Represents a relationship between two fulfillment order lines. This object is available in API version 61.0 and later.

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.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`AssociatedFoItemInventory`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort

Field	Details
	Description Inventory level for the associated fulfillment order item. Valid values are: • `Included in Main Inventory` • `Not Included in Main Inventory` This field is available in API version 63.0 and later.
`AssociatedFulfillOrderItemId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The identifier of the associated fulfillment order line item. This field is a relationship field. **Relationship Name** AssociatedFulfillOrderItem **Refers To** FulfillmentOrderLineItem
`AssociatedLineRole`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The role of the associated fulfillment order line item. Valid values are: • `BundleComponent` • `ClassificationComponent`—Product Classification Component
`AssociatedQuanScaleMethod`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Method used to scale the quantity of the associated order item summary relative to the main fulfillment order item. Valid values are: • `Constant` • `Proportional`

Field	Details
	The default value is `Proportional`. This field is available in API version 63.0 and later.
`FulfillmentOrderId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The ID of the parent fulfillment order. This field is a relationship field. **Relationship Name** FulfillmentOrder **Refers To** FulfillmentOrder
`MainFulfillOrderItemRole`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The role of the primary fulfillment order line item. Valid value is `Bundle`.
`MainFulfillmentOrderItemId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The ID of the associated fulfillment order line item. This field is a relationship field. **Relationship Name** MainFulfillmentOrderItem **Relationship Type** Master-detail **Refers To** FulfillmentOrderLineItem (the master object)
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort

Field	Details
	Description The name of the fulfillment order line relationship.
`ProductRelationshipTypeId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The type of relationship between two assets. This field is a relationship field. **Relationship Name** ProductRelationshipType **Refers To** ProductRelationshipType

FulfillmentLineSourceRel

Represents the relationship between a fulfillment order line and its decomposition source. This object is available in API version 61.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`FulfilmentOrderLineId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The ID of the fulfillment order line. This field is a relationship field. **Relationship Name** FulfilmentOrderLine **Refers To** FulfillmentOrderLineItem

Field	Details
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The relation between the two order line sources.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The ID of the user who created the relationship record. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
`SourceLineItemId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The ID of the source line item. This field is a polymorphic relationship field. **Relationship Name** SourceLineItem **Refers To** FulfillmentOrderLineItem, OrderItem
`SourceType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The type of source for the line item. Valid values are: • `SourceBundleRoot` • `SourceLineItem`

Field	Details
`SupplementalAction`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort **Description** The supplemental action that's applied to the line item based on the run-time changes made to the original fulfillment request. This field is available in API version 62.0 and later. Valid values are: • `Add` • `Amend` • `Cancel` • `NoChange`
`UniqueIdentifier`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort **Description** For internal use only.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

FulfillmentLineSourceRelShare **on page 1684**
Sharing is available for the object.

FulfillmentOrder

Represents a group of products, fees, and delivery charges on a single order that share the same fulfillment location, delivery method, and recipient. The FulfillmentOrderLineItems belonging to a FulfillmentOrder are associated with OrderItemSummary objects belonging to a single OrderSummary. This object is available in API version 48.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

This object is only available in Salesforce Order Management orgs.

Fields

Field	Details
`AccountId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort **Description** ID of the Account or Person Account associated with the FulfillmentOrder. It represents the shopper in the storefront. This field is a relationship field. **Relationship Name** Account **Relationship Type** Lookup **Refers To** Account
`ActiveDate`	**Type** datetime **Properties** Filter, Group, Nillable, Sort **Description** Date when the fulfillment order becomes active. This field is available in API version 61.0 and later.
`BillToContactId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort **Description** ID of the Contact associated with the FulfillmentOrder. It represents the shopper in the storefront when not using person accounts. This field is available in API version 49.0 and later. This field is a relationship field. **Relationship Name** BillToContact **Relationship Type** Lookup **Refers To** Contact

Field	Details
`ClosedDate`	**Type** datetime **Properties** Filter, Group, Nillable, Sort **Description** Date the fulfillment order closed. Automatically entered. This field is available in API version 61.0 and later.
`CurrencyIsoCode`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Available only for orgs with the multicurrency feature enabled. ISO code for the currency of the OrderSummary associated with the FulfillmentOrder. Possible values are: • `DKK`—Danish Krone • `EUR`—Euro • `GBP`—British Pound • `USD`—U.S. Dollar The default value is `USD`. This field is available in API version 49.0 and later.
`DeliveryMethodId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** ID of the DeliveryMethod used for this FulfillmentOrder. This field is a relationship field. **Relationship Name** DeliveryMethod **Relationship Type** Lookup **Refers To** OrderDeliveryMethod
`FulfilledFromLocationId`	**Type** reference

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update **Description** ID of the Location handling this FulfillmentOrder. This field is a relationship field. **Relationship Name** FulfilledFromLocation **Relationship Type** Lookup **Refers To** Location
`FulfilledToAddress`	**Type** address **Properties** Filter, Nillable **Description** Address of the recipient.
`FulfilledToCity`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Recipient address city.
`FulfilledToCountry`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Recipient address country.
`FulfilledToEmailAddress`	**Type** email **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Email address of the recipient.

Field	Details
`FulfilledTo GeocodeAccuracy`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Accuracy of the geocode for the recipient address. Possible values are: • `Address` • `Block` • `City` • `County` • `ExtendedZip` • `NearAddress` • `Neighborhood` • `State` • `Street` • `Unknown` • `Zip`
`FulfilledToLatitude`	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** Used with FulfilledToLongitude to specify the precise geolocation of the recipient address. Acceptable values are numbers between –90 and 90 with up to 15 decimal places.
`FulfilledToLongitude`	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** Used with FulfilledToLatitude to specify the precise geolocation of the recipient address. Acceptable values are numbers between –90 and 90 with up to 15 decimal places.
`FulfilledToName`	**Type** string **Properties** Create, Filter, Group, Sort, Update

Field	Details
	Description Name on the recipient address.
`FulfilledToPhone`	**Type** phone **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Phone number of the recipient.
`FulfilledToPostalCode`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Recipient address postal code.
`FulfilledToState`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Recipient address state.
`FulfilledToStreet`	**Type** textarea **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Recipient address street.
`FulfillmentOrderNumber`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** ID of the FulfillmentOrder.
`GrandTotalAmount`	**Type** currency **Properties** Filter, Nillable, Sort

Field	Details
	Description Total, including adjustments and tax, of the products, fees, and delivery charges on the FulfillmentOrder. This amount includes all FulfillmentOrderLineItems associated with the FulfillmentOrder. This amount is equal to TotalAmount + TotalTaxAmount.
`InvoiceId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** ID of the Invoice associated with the FulfillmentOrder. This field is a relationship field. **Relationship Name** Invoice **Relationship Type** Lookup **Refers To** Invoice
`IsReship`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the FulfillmentOrder is for a reshipment. The default value is false. This field is available in API version 53.0 and later.
`IsSuspended`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the FulfillmentOrder is suspended. The default value is false.
`ItemCount`	**Type** double **Properties** Filter, Nillable, Sort **Description** Sum of the quantities of the FulfillmentOrderLineItems included in the FulfillmentOrder.

Field	Details
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Timestamp for when the current user last viewed a record related to this record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Timestamp for when the current user last viewed this record. A null value can mean that this record has only been referenced (LastReferencedDate) and not viewed.
`OrderId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort **Description** ID of the original Order that generated the FulfillmentOrder. This field is a relationship field. **Relationship Name** Order **Relationship Type** Lookup **Refers To** Order
`OrderSummaryId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort **Description** ID of the OrderSummary associated with the FulfillmentOrder. This field is a relationship field. **Relationship Name** OrderSummary **Relationship Type** Lookup

Field	Details
	Refers To OrderSummary
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** ID of the User who currently owns this FulfillmentOrder. Default value is the User logged in to the API to perform the create. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User
`Status`	**Type** picklist **Properties** Create, Filter, Group, Sort, Update **Description** Status of the FulfillmentOrder. Each status corresponds to one status category, shown here in parentheses. You can customize the status picklist to represent your business processes, but the status category picklist is fixed because processing is based on those values. If you customize the status picklist, include at least one status value for each status category. Default values are: • `Allocated` (Activated) • `Assigned` (Fulfilling) • `Cancelled` (Cancelled) • `Draft` (Draft) • `Fulfilled` (Closed) • `Pick Complete` (Fulfilling) This value is available in API v56.0 and later. • `Pickpack` (Fulfilling) • `Printed` (Fulfilling) This value is available in API v56.0 and later. • `Rejected` (Rejected) This value is available in API v56.0 and later.
`StatusCategory`	**Type** picklist

Field	Details
	Properties Filter, Group, Restricted picklist, Sort **Description** Status category of the FulfillmentOrder. Processing of the FulfillmentOrder depends on this value. Each status category corresponds to one or more statuses. Possible values are: • `ACTIVATED`—Activated • `CANCELLED`—Cancelled • `CLOSED`—Closed • `DRAFT`—Draft • `FULFILLING`—Fulfilling • `REJECTED`—Rejected This value is available in API v56.0 and later.
`TaxLocaleType`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort **Description** The system used to handle tax on the original Order associated with the FulfillmentOrder. Gross usually applies to taxes like value-added tax (VAT), and Net usually applies to taxes like sales tax. Possible values are: • `Gross` (displays most prices and taxes as combined values) • `Net` (displays most prices and taxes as separate values) This field is available in API version 49.0 and later.
`TotalAdjustmentAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Total amount of the price adjustments applied to the products on the FulfillmentOrder. This value only includes adjustments to FulfillmentOrderLineItems of type code Product, not adjustments to delivery charges or fees.
`TotalAdjustment AmtWithTax`	**Type** currency **Properties** Filter, Nillable, Sort

Field	Details
	Description Total amount of the price adjustments applied to the products on the FulfillmentOrder, inclusive of tax. This value only includes adjustments to FulfillmentOrderLineItems of type code Product. This amount is equal to TotalAdjustmentAmount + TotalAdjustmentTaxAmount. This field is available in API version 49.0 and later.
`TotalAdjustmentTaxAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Tax on the TotalAdjustmentAmount.
`TotalAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Adjusted total, not including tax, of the FulfillmentOrderLineItems, including products, fees, and delivery charges, on the FulfillmentOrder.
`TotalDelivery AdjustAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Total amount of the price adjustments applied to the delivery charges on the FulfillmentOrder. This value only includes adjustments to FulfillmentOrderLineItems of type Delivery Charge.
`TotalDeliveryAdjust AmtWithTax`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Total amount of the price adjustments applied to the delivery charges on the FulfillmentOrder, inclusive of tax. This value only includes adjustments to FulfillmentOrderLineItems of type Delivery Charge. This amount is equal to TotalDeliveryAdjustAmount + TotalDeliveryAdjustTaxAmount. This field is available in API version 49.0 and later.

Field	Details
`TotalDeliveryAdjustTaxAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Tax on the TotalDeliveryAdjustAmount.
`TotalDeliveryAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Total of the delivery charges on the FulfillmentOrder. This value only includes FulfillmentOrderLineItems of type Delivery Charge.
`TotalDeliveryAmtWithTax`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Total amount of the delivery charges on the FulfillmentOrder, inclusive of tax. This value only includes FulfillmentOrderLineItems of type Delivery Charge. This amount is equal to TotalDeliveryAmount + TotalDeliveryTaxAmount. This field is available in API version 49.0 and later.
`TotalDeliveryTaxAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Tax on the TotalDeliveryAmount.
`TotalFeeAdjustAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Total amount of the price adjustments applied to the fees on the FulfillmentOrder. This value only includes adjustments to FulfillmentOrderLineItems of type Fee. This field is available in API version 56.0 and later.

Field	Details
`TotalFeeAdjustAmtWithTax`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Total amount of the price adjustments applied to the fees on the FulfillmentOrder, inclusive of tax. This value only includes adjustments to FulfillmentOrderLineItems of type Fee. This amount is equal to TotalFeeAdjustAmount + TotalFeeAdjustTaxAmount. This field is available in API version 56.0 and later.
`TotalFeeAdjustTaxAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Tax on the TotalFeeAdjustAmount. This field is available in API version 56.0 and later.
`TotalFeeAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Total amount of the fees on the FulfillmentOrder, excluding adjustments and tax. This value only includes FulfillmentOrderLineItems of type Fee. This field is available in API version 56.0 and later.
`TotalFeeAmtWithTax`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Total price of the fees on the FulfillmentOrder, inclusive of tax. This value only includes FulfillmentOrderLineItems of type Fee. This amount is equal to TotalFeeAmount + TotalFeeTaxAmount. This field is available in API version 56.0 and later.
`TotalFeeTaxAmount`	**Type** currency **Properties** Filter, Nillable, Sort

Field	Details
	Description Tax on the TotalFeeAmount. This field is available in API version 56.0 and later.
`TotalProductAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Total price of the products on the FulfillmentOrder, excluding order adjustments, delivery charges, and fees. This value only includes FulfillmentOrderLineItems of type code Product.
`TotalProductAmtWithTax`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Total price of the products on the FulfillmentOrder, inclusive of tax. This value only includes FulfillmentOrderLineItems of type code Product. This amount is equal to TotalProductAmount + TotalProductTaxAmount. This field is available in API version 49.0 and later.
`TotalProductTaxAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Tax on the TotalProductAmount.
`TotalTaxAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Tax on the TotalAmount.
`Type`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description Type of the FulfillmentOrder. Each type corresponds to one type category, shown here in parentheses. You can customize the type picklist to represent your business processes, but the type category picklist is fixed because processing is based on those values. If you customize the type picklist, include at least one type value for each type category. Default values are: • `Download` (Digital) • `Email` (Digital) • `In Store Pickup` (Physical) • `Retail Store` (Physical) • `Supplier` (Drop Ship) • `Warehouse` (Physical)
`TypeCategory`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** Type category of the FulfillmentOrder. Processing of the FulfillmentOrder depends on this value. Each type category corresponds to one or more types. Possible values are: • `DIGITAL`—Digital • `DROPSHIP`—Drop Ship • `PHYSICAL`—Physical

Associated Objects

This object has the following associated objects. Unless noted, they are available in the same API version as this object.

FulFillmentOrderChangeEvent (API version 62.0)
Change events are available for the object.

FulfillmentOrderFeed
Feed tracking is available for the object.

FulfillmentOrderOwnerSharingRule
Sharing rules are available for the object.

FulfillmentOrderShare
Sharing is available for the object.

FulfillmentOrderLineItem

Represents a product or delivery charge belonging to a FulfillmentOrder. Corresponds to an OrderItemSummary. This object is available in API version 48.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

This object is only available in Salesforce Order Management orgs.

Fields

Field	Details
`CurrencyIsoCode`	**Type** picklist **Properties** Defaulted on create, Filter, Group, Restricted picklist, Sort **Description** Available only for orgs with the multicurrency feature enabled. ISO code for the currency of the FulfillmentOrder associated with the FulfillmentOrderLineItem. Possible values are: • `DKK`—Danish Krone • `EUR`—Euro • `GBP`—British Pound • `USD`—U.S. Dollar The default value is `USD`. This field is available in API version 49.0 and later.
`Description`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Description of the FulfillmentOrderLineItem.
`EndDate`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort

Field	Details
	Description End date of the FulfillmentOrderLineItem.
`FulfillmentOrderId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** ID of the FulfillmentOrder associated with the FulfillmentOrderLineItem. This field is a relationship field. **Relationship Name** FulfillmentOrder **Relationship Type** Lookup **Refers To** FulfillmentOrder
`FulfillmentOrder LineItemNumber`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** ID of the FulfillmentOrderLineItem.
`GrossUnitPrice`	**Type** currency **Properties** Create, Filter, Nillable, Sort **Description** Unit price, including tax, of the FulfillmentOrderLineItem. This value is equal to TotalPrice + TotalTaxAmount. This field is available in API version 49.0 and later.
`IsReship`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Indicates whether the FulfillmentOrderLineItem belongs to a reshipment. The default value is false. This field is available in API version 53.0 and later.

Field	Details
MainFulfillmentOrderLineItemId	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort **Description** ID of the original FulfillmentOrderLineItem. This field is a relationship field. This field is available in API version 63.0 and later. **Relationship Name** FulfillmentOrderLineItem **Relationship Type** Lookup **Refers To** FulfillmentOrderLineItem
OrderItemId	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort **Description** ID of the original OrderItem for the OrderItemSummary associated with the FulfillmentOrderLineItem. This field is a relationship field. **Relationship Name** OrderItem **Relationship Type** Lookup **Refers To** OrderItem
OrderItemSummaryId	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort **Description** ID of the OrderItemSummary associated with the FulfillmentOrderLineItem. This field is a relationship field. **Relationship Name** OrderItemSummary

Field	Details
	Relationship Type Lookup **Refers To** OrderItemSummary
`OriginalQuantity`	**Type** double **Properties** Create, Filter, Sort **Description** Original quantity of the FulfillmentOrderLineItem.
`Product2Id`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort **Description** ID of the product represented by the FulfillmentOrderLineItem. This field is a relationship field. **Relationship Name** Product2 **Relationship Type** Lookup **Refers To** Product2
`Quantity`	**Type** double **Properties** Create, Filter, Sort, Update **Description** Current quantity of the FulfillmentOrderLineItem. Equal to the original quantity minus any canceled quantity.
`QuantityUnitOfMeasure`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort **Description** Unit of measure of the quantity, for example: unit, gallon, ton, or case.

Field	Details
`RejectedQuantity`	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** Used by the Distributed Order Management package and Store Fulfillment app to store the quantity that has been rejected by a fulfillment location. This field is available in API version 57.0 and later.
`RejectReason`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** If the FulfillmentOrderLineItem was rejected by a fulfillment location, the reason for the rejection. Default values are: • `Damaged` • `Just Sold` • `Other` • `Out of Packing Supplies` • `Out of Stock` This field is available in API version 56.0 and later.
`ReshipReason`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** If the FulfillmentOrderLineItem belongs to a reshipment, the reason for the reshipment. Default values are: • `Damaged` • `Lost` • `Unknown` • `Wrong Item` This field is available in API version 53.0 and later.
`ServiceDate`	**Type** dateTime

Field	Details
	Properties Create, Filter, Nillable, Sort **Description** Service or start date of the FulfillmentOrderLineItem.
`TotalAdjustmentAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Total of any price adjustments applied to the FulfillmentOrderLineItem.
`TotalAdjustment AmountWithTax`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Total amount of the price adjustments applied to the FulfillmentOrderLineItem, inclusive of tax. This amount is equal to TotalAdjustmentAmount + TotalAdjustmentTaxAmount. This field is available in API version 49.0 and later.
`TotalAdjustment TaxAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Tax on the TotalAdjustmentAmount.
`TotalAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Total, including adjustments and tax, of the FulfillmentOrderLineItem.
`TotalLineAmount`	**Type** currency **Properties** Create, Defaulted on create, Filter, Nillable, Sort **Description** Total, not including adjustments or tax, of the FulfillmentOrderLineItem.

Field	Details
`TotalLineAmountWithTax`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Total price of the FulfillmentOrderLineItem, inclusive of tax. This amount is equal to TotalLineAmount + TotalLineTaxAmount. This field is available in API version 49.0 and later.
`TotalLineTaxAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Tax on the TotalLineAmount.
`TotalPrice`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Total, including adjustments but not tax, of the FulfillmentOrderLineItem. Equal to UnitPrice times Quantity.
`TotalTaxAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Tax on the TotalPrice.
`Type`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort **Description** Type of the FulfillmentOrderLineItem. Matches the type of the associated OrderItemSummary. Delivery Charge indicates that the FulfillmentOrderLineItem represents a delivery charge. Fee indicates that it represents another type of fee, such as a return fee. Order Product indicates that it represents any other type of product, service, or charge. Each type corresponds to one type code, shown here in parentheses. Possible values are:

Field	Details
	• `Delivery Charge` (Charge) • `Fee` (Charge) This value is available in API v56.0 and later. • `Order Product` (Product)
`TypeCode`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort **Description** Type code of the FulfillmentOrderLineItem. Matches the type code of the associated OrderItemSummary. Processing depends on this value. Charge indicates that the FulfillmentOrderLineItem represents a charge or fee. Product indicates that it represents any other type of product, service, or charge. A type code can be associated with one or more types. Possible values are: • `Charge` • `Product`
`UnitPrice`	**Type** currency **Properties** Create, Defaulted on create, Filter, Nillable, Sort **Description** Unit price of the FulfillmentOrderLineItem.

FulFillmentOrderChangeEvent (API version 62.0)

Change events are available for the object.

FulfillmentPlan

Represents a set of steps to be created to fulfill the order. This object is available in API version 61.0 and later.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), undelete(), update(), upsert()`

Fields

Field	Details
ExecutionUserId	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** For internal use only. **Relationship Name** ExecutionUser **Refers To** User
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The most recent date when a user referenced this record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The most recent date when a user viewed this record.
Name	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name of the fulfillment plan.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The ID of the user who created the record. This field is a polymorphic relationship field.

Field	Details
	Relationship Name Owner **Refers To** Group, User
`Priority`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** The priority of the fulfillment plan execution. This field is available in API version 63.0 and later. Valid values are: • `Default` • `High` • `Bulk`
`SourceIdentifier`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** For internal use only.
`SourceType`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort **Description** The type of source for the fulfillment plan. This field is available in API version 62.0 and later.
`State`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The status of the fulfillment plan. Valid values are: • `Completed` • `InProgress`

Field	Details
	• `NotStarted`
`UsageType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The business that uses Fulfillment Orchestration. Valid values are: • `IntegrationOrchestrator` • `OrderFulfillment` • `StageManagement`

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

FulfillmentPlanChangeEvent on page 1671
Change events are available for the object.

FulfillmentPlanShare on page 1684
Sharing is available for the object.

FulfillmentStep

Represents a task that's required to perform a certain action as part of order fulfillment. This task can be manual or automated. This object is available in API version 61.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`ActualCompletionDate`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update

Field	Details
	Description The date and time when the fulfillment step state changed to `Completed` or `Skipped`.
`ActualStartDate`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date and time when the fulfillment step state changed to `Ready`.
`AssignedToId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The user or queue associated with the fulfillment step. This field is a polymorphic relationship field. **Relationship Name** AssignedTo **Relationship Type** Lookup **Refers To** Group, User
`CompensatedStepId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort **Description** The alternative step that's executed when a particular step in the fulfillment plan is amended or canceled. This field is a relationship field. This field is available in API version 62.0 and later. **Relationship Name** CompensatedStep **Refers To** FulfillmentStep
`DelayOf`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The value of the delay. This field is available in API version 63.0 and later.
`DelayUnit`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The unit for the delay of the fulfillment step. This field is available in API version 63.0 and later. Valid values are: • `Days` • `Hours` • `Minutes`
`ExecuteOn`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Specifies when to execute the fulfillment step. This field is available in API version 63.0 and later. Valid values are: • `PreviousStepExecutionDate` • `SourceLineStartDate`
`ExecuteOnRuleId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The name of the expression set. The fulfillment step is executed only when its corresponding expression set returns the value `true`. This field is a polymorphic relationship field. **Relationship Name** ExecuteOnRule **Relationship Type** Lookup **Refers To** ExpressionSet

Field	Details
`ExecutionMessage`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** For internal use only.
`FalloutQueueId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The fallout queue that's associated with the fallout task. This field is available in API version 62.0 and later. This field is a relationship field. **Relationship Name** FalloutQueue **Refers To** Group
`FlowDefinitionName`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The name of the flow definition that's associated with the `AutoTask` type of fulfillment step. This field is available in API version 62.0 and later.
`FlowInterviewId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The flow interview associated with the fulfillment step. This field is a relationship field. **Relationship Name** FlowInterview **Relationship Type** Lookup **Refers To** FlowInterview

Field	Details
`ForcePlanFreezeDuringExecution`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Specifies whether to freeze the plan while the step is in progress. If enabled, specifies how to complete the step before resuming the plan. This field is available in API version 63.0 and later. Valid values are: • `Never` • `YesButWaitForStepCompletion` • `YesButForcefullyCompleteStep` The default value is `Never`.
`FulfillmentPlanId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The fulfillment plan associated with the fulfillment step. This field is a relationship field. **Relationship Name** FulfillmentPlan **Relationship Type** Lookup **Refers To** FulfillmentPlan
`FulfillmentStepDefinitionId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The fulfillment step definition associated with the fulfillment step. This field is a relationship field. **Relationship Name** FulfillmentStepDefinition **Relationship Type** Lookup

Field	Details
	Refers To FulfillmentStepDefinition
`IntegrationDefinitionNameId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** For internal use only. This field is a relationship field. **Relationship Name** IntegrationDefinitionName **Relationship Type** Lookup **Refers To** IntegrationProviderDef
`IsSkipBranch`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the remaining steps in the fulfillment step group are skipped from execution when the Execute On Rule condition is set (`true`) or not (`false`). This field is available in API version 62.0 and later. The default value is `false`.
`JeopardyStatus`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The jeopardy status of the fulfillment step. This field is a calculated field.
`JeopardyThreshold`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The number of days, hours, minutes, or seconds, counting back from the expected duration, before a fulfillment step is in jeopardy.
`JeopardyThresholdUnit`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The time unit of the jeopardy threshold. Valid values are: • `Days` • `Hours` • `Minutes` • `Seconds` The default value is `Minutes`.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The most recent date when a user referenced this record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The most recent date when a user viewed this record.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** The name of the fulfillment step.
`NextEarliestRunTime`	**Type** dateTime

Field	Details
	Properties Create, Filter, Nillable, Sort, Update **Description** The next available time for the process execution. This field is available in API version 62.0 and later.
`OmniscriptName`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** For inyternal use only.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The ID of the user who created the record. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User
`PlannedCompletionDate`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date and time Dynamic Revenue Orchestrator estimates that the fulfillment step will complete.
`PlannedStartDate`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update

Field	Details
	Description The date and time when the Dynamic Revenue Orchestrator calculates the fulfillment step state change to `Ready`.
`PointOfNoReturn`	**Type** multipicklist **Properties** Create, Filter, Nillable **Description** The type of source change applied to the line item. This field is available in API version 62.0 and later. Valid value is: • `Changes Denied`
`RequestedCompletionDate`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The planned completion date and time of the fulfillment step.
`RequestedStartDate`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The planned start date and time of the fulfillment step.
`ResumeOnRuleId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The rule set or expression set for the fulfillment step. The step is completed only when the corresponding expression set returns the `isExecuteStep` output as `true`. This field is a polymorphic relationship field. **Relationship Name** ResumeOnRule **Relationship Type** Lookup

Field	Details
	Refers To ExpressionSet
`RetryAttempts`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The number of attempts allowed for retry as set up in the Fallout Qualification Rule table.
`RunAsUserId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Overrides user context for an automated step. The default user context is AutomatedProc user. This field is a relationship field. **Relationship Name** RunAsUser **Relationship Type** Lookup **Refers To** User
`State`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The state of the fulfillment step. For example, `In Progress` or `Completed`. Valid values are: • `Completed` • `Failed` • `FatallyFailed` • `InProgress` • `Pending` • `Ready` • `Skipped`

Field	Details
`StepType`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The fulfillment step type associated with the fulfillment step. Valid values are: • `AutoTask` • `Callout` • `ManualTask` • `Milestone` • `Pause` • `StagedAssetize`—Available in API version 63.0 and later.
`TaskAllocationType`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The method of assigning the manual step. This field is available in API version 63.0 and later. Valid values are: • `ContextBased` • `LeastLoaded` • `RoundRobin`
`TaskId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ID of the task assigned to a user or queue. This field is available in API version 63.0 and later. This field is a relationship field. **Relationship Name** Task **Refers To** Task
`UsageType`	**Type** picklist

Field	Details
	Properties Create, Filter, Group, Restricted picklist, Sort, Update **Description** The details about the business that uses Fulfillment Orchestration. Some examples of UsageBy include Financial Services Cloud and CPQ. Valid values are: • `Fulfillment` • `InsuranceRuleAction` • `IntegrationOrchestrator` • `OrderFulfillment` • `StageManagement`

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

FulfillmentStepChangeEvent on page 1671
Change events are available for the object.

FulfillmentStepHistory on page 1680
History is available for tracked fields of the object.

FulfillmentStepShare on page 1684
Sharing is available for the object.

FulfillmentStepDefinition

Represents a definition of a step that must be executed during fulfillment orchestration. This object is available in API version 61.0 and later.

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.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`AmendGroupId`	**Type** reference

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update **Description** The fulfillment step group that's added to the fulfillment plan when the step is amended. This field is available in API version 62.0 and later. This field is a relationship field. **Relationship Name** AmendGroup **Refers To** FulfillmentStepDefinitionGroup
`AssignedToId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The user or queue associated with the fulfillment step definition. This field is a polymorphic relationship field. **Relationship Name** AssignedTo **Relationship Type** Lookup **Refers To** Group, User
`CancelledGroupId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The fulfillment step group that's added to the fulfillment plan when the step is canceled. This field is available in API version 62.0 and later. This field is a relationship field. **Relationship Name** CancelledGroup **Refers To** FulfillmentStepDefinitionGroup
`DelayOf`	**Type** int

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update **Description** The value of the delay. This field is available in API version 63.0 and later.
`DelayUnit`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The unit for the delay of the fulfillment step. This field is available in API version 63.0 and later. Valid values are: • `Days` • `Hours` • `Minutes`
`ExecuteOn`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Specifies when to execute the fulfillment step. This field is available in API version 63.0 and later. Valid values are: • `PreviousStepsStartDate` • `SourceLineStartDate`
`ExecuteOnRuleId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The expression set for the fulfillment step. The step is executed only when the corresponding expression set is `true`. This field is a polymorphic relationship field. **Relationship Name** ExecuteOnRule **Relationship Type** Lookup

Field	Details
	Refers To ExpressionSet
`FlowDefinitionName`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The name of the associated flow.
`ForcePlanFreezeDuringExecution`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Specifies whether to freeze the plan while the step is in progress. If enabled, specifies how to complete the step before resuming the plan. This field is available in API version 63.0 and later. Valid values are: • `Never` • `YesButForcefullyCompleteStep` The default value is `Never`.
`IntegrationDefinitionNameId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The name of the integration definition that's used to set up communication with an external endpoint. This field is a relationship field. **Relationship Name** IntegrationDefinitionName **Relationship Type** Lookup **Refers To** IntegrationProviderDef
`IsSkipBranch`	**Type** boolean

Field	Details
	Properties Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the remaining steps in the fulfillment step group are skipped from execution when the Execute On Rule condition is set. This field is available in API version 62.0 and later. The default value is `false`.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The most recent date when a user referenced this record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The most recent date when a user viewed this record.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** The name of the fulfillment step definition.
`OmniscriptName`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** For internal use only.
`PointOfNoReturn`	**Type** multipicklist **Properties** Create, Filter, Nillable, Update

Field	Details
	Description The type of source change applied to the line item. This field is available in API version 62.0 and later. Valid value is: • `Changes Denied`
`ResumeOnRuleId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The expression set for the fulfillment step definition. The step is completed when the expression set is `true`. This field is a polymorphic relationship field. **Relationship Name** ResumeOnRule **Relationship Type** Lookup **Refers To** ExpressionSet
`RunAsUserId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The user context to run a fulfillment step when the fulfillment operator wants to override the default autoproc user. This field is a relationship field. **Relationship Name** RunAsUser **Relationship Type** Lookup **Refers To** User
`Scope`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field	Details
	Description The scope of the fulfillment step definition. For example, `Bundle` or `Order`. Valid values are: • `Bundle` • `LineItem` • `Plan` The default value is `Plan`.
`StepDefinitionGroupId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** For internal use only. This field is a relationship field. **Relationship Name** StepDefinitionGroup **Relationship Type** Master-detail **Refers To** FulfillmentStepDefinitionGroup (the master object)
`StepType`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** For internal use only. Valid values are: • `AutoTask` • `Callout` • `ManualTask` • `Milestone` • `Pause`
`TaskAllocationType`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field	Details
	Description The method of assigning the manual step. This field is available in API version 62.0 and later. Valid value is: • `RoundRobin`
`UsageType`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The business vertical that uses fulfillment orchestration. For example, Financial Services Cloud. Valid values are: • `IntegrationOrchestrator` • `OrderFulfillment` • `StageManagement`

FulfillmentStepDefinitionGroup

Represents a set of fulfillment step definitions. This object is available in API version 61.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The most recent date when a user referenced this record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort

Field	Details
	Description The most recent date when a user viewed this record.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** The name of the fulfillment step definition group.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The ID of the user who created the record. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
`UsageType`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The business that uses Fulfillment Orchestration. Possible values are: • `IntegrationOrchestrator` • `OrderFulfillment` • `StageManagement`

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

FulfillmentStepDefinitionGroupShare **on page 1684**
Sharing is available for the object.

FulfillmentStepDependency

Represents a dependency between tasks by defining the order between a task and one that depends on it. This object is available in API version 61.0 and later.

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.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`DependencyDefinitionId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The name of the fulfillment step dependency definition. This field is a relationship field. **Relationship Name** DependencyDefinition **Refers To** FulfillmentStepDependencyDef
`DependentStepId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The name of a fulfillment step that depends on this step. This field is a relationship field. **Relationship Name** DependentStep **Relationship Type** Master-detail **Refers To** FulfillmentStep

Field	Details
DependsOnStepId	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The name of the fulfillment step that this step depends on. That is, the name of the step that must be executed before this one can run. This field is a relationship field. **Relationship Name** DependsOnStep **Refers To** FulfillmentStep
Name	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** An automatically generated name for the fulfillment step dependency.

FulfillmentStepDependencyDef

Represents a dependency that must be created between two fulfillment step records. This object is available in API version 62.0 and later.

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.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
DependencyScope	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update

Field	Details
	Description The scope of the fulfillment step dependency definition. For example, Order or Order Item. Valid values are: • `Bundle` • `LineItem` • `Plan` The default value is `Plan`.
`DependsOnStepDefinitionId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The fulfillment step definition that must be executed before this step. This field is a relationship field. **Relationship Name** DependsOnStepDefinition **Refers To** FulfillmentStepDefinition
`FulfillmentStepDefinitionId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The identifier of the fulfillment step definition. This field is a relationship field. **Relationship Name** FulfillmentStepDefinition **Relationship Type** Master-detail **Refers To** FulfillmentStepDefinition (the master object)
`IsCompensateInReverse`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update

Field	Details
	Description Indicates whether the order to insert the compensated group steps is reversed when a fulfillment step is canceled (`true`) or not (`false`). The default value is `false`. This field is available in API version 63.0 and later.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** The name of the fulfillment step dependency definition.
`PropagateStateToDependentStep`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The state that's propagated to the dependent fulfillment step when the source fulfillment step is amended or canceled in the fulfillment plan. Valid values are: • `Amended` • `Both` • `Canceled` • `None` This field is available in API version 63.0 and later.

FulfillmentStepJeopardyRule

Represents the duration and tolerance for the step in the fulfillment process to allow the overall tracking of rules and risks. This object is available in API version 61.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`EstimatedDuration`	**Type** int **Properties** Create, Filter, Group, Sort, Update **Description** The estimated time to complete the fulfillment step.
`EstimatedDurationUnit`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update **Description** The unit of measurement that applies to the estimated time to complete the fulfillment step. Valid values are: • `Days` • `Hours` • `Minutes` • `Seconds` The default value is `Minutes`.
`FlowDefinition`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The flow definition that's associated with the `AutoTask` type of the fulfillment step.
`IntegrationDefinitionId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The integration definition that's associated with the `Callout` type of the fulfillment step. This field is a relationship field. **Relationship Name** IntegrationDefinition **Relationship Type** Lookup

Field	Details
	Refers To IntegrationProviderDef
`JeopardyThreshold`	**Type** int **Properties** Create, Filter, Group, Sort, Update **Description** The value indicating the threshold after which the fulfillment step is in jeopardy.
`JeopardyThresholdUnit`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update **Description** The unit of measurement that applies to the jeopardy threshold value. Valid values are: • `Days` • `Hours` • `Minutes` • `Seconds` The default value is `Minutes`.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** For internal use only.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** For internal use only.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort

Field	Details
	Description For internal use only.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** For internal use only. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User
`StepType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The type of fulfillment step that's affected by the jeopardy rule. Valid values are: • `AutoTask` • `Callout` • `ManualTask` • `Milestone` • `Pause`

FulfillmentStepSource

Represents a link between a fulfillment step and the corresponding order lines. This object is available in API version 61.0 and later.

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.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** An automatically generated name for the fulfillment step source.
`SourceIdentifier`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The identifier of the source order line item (order item or fulfillment order line).
`SourceLineItemId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** For internal use only. This field is a polymorphic relationship field. **Relationship Name** SourceLineItem **Refers To** FulfillmentOrderLineItem, OrderItem
`StepId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The identifier of the fulfillment step. This field is a relationship field. **Relationship Name** Step **Relationship Type** Master-detail **Refers To** FulfillmentStep (the master object)

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

FulfillmentStepSourceChangeEvent on page 1671
Change events are available for the object.

FulfillmentTaskAssignmentRule

Represents a set of actions that assign a task to a user or queue. This object is available in API version 63.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`ConditionId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Condition ID that's used to determine the task assignment. This field is a polymorphic relationship field. **Relationship Name** Condition **Refers To** ExpressionSet
`DestinationId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** Destination ID of the task assignment such as, Queue or User. This field is a polymorphic relationship field. **Relationship Name** Destination **Refers To** Group, User

Field	Details
LastReferencedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The most recent date when a user referenced this record.
LastViewedDate	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The most recent date when a user viewed this record.
Name	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** The name of the object that specifies the condition used to determine the task assignment.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The ID of the user who created the task assignment rule record. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
Priority	**Type** int **Properties** Create, Filter, Group, Sort, Update **Description** The priority of the rule for execution.

Field	Details
`SourceId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** Source ID of the task assignment such as, Queue or Fulfillment Step Definition. This field is a relationship field. **Relationship Name** Source **Refers To** Group
`TaskAllocationType`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The method of assigning the manual step. Valid values are: • `ContextBased` • `LeastLoaded` • `RoundRobin`

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

FulfillmentTaskAssignmentRuleFeed
Feed tracking is available for the object.

FulfillmentTaskAssignmentRuleHistory
History is available for tracked fields of the object.

FulfillmentTaskAssignmentRuleShare
Sharing is available for the object.

FulfillmentWorkspace

Represents a visual designer for fulfillment plans that can have multiple step groups and their dependencies. This object is available in API version 61.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Description`	**Type** textarea **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The description of the fulfillment workspace.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The most recent date when a user referenced this record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The most recent date when a user viewed this record.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** The name of the fulfillment workspace.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The ID of the user who owns this record. This field is a polymorphic relationship field.

Field	Details
	Relationship Name Owner **Refers To** Group, User

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

FulfillmentWorkspaceShare **on page 1684**
Sharing is available for the object.

FulfillmentWorkspaceItem

Represents information about the attributes that are used in the definition for a fulfillment step group. This object is available in API version 61.0 and later.

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.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`FulfillmentStepDefinitionGroupId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** For internal use only. This field is a relationship field. **Relationship Name** FulfillmentStepDefinitionGroup **Relationship Type** Master-detail **Refers To** FulfillmentStepDefinitionGroup (the detail object)

Field	Details
`FulfillmentWorkspaceId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The ID of the parent fulfillment workspace that's related to this record. This field is a relationship field. **Relationship Name** FulfillmentWorkspace **Relationship Type** Master-detail **Refers To** FulfillmentWorkspace (the master object)
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name of the fulfillment workspace item.
`ShowOrder`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The display sequence value of the fulfillment workspace item.

Order

Represents an order associated with a contract or an account.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field Name	Details
`AccountId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Required. ID of the Account associated with this order. Can only be updated when the order's `StatusCode` value is *`Draft`*. This is a relationship field. **Relationship Name** Account **Relationship Type** Lookup **Refers To** Account
`ActivatedById`	**Type** reference **Properties** Filter, Group, Nillable, Sort, Update **Description** ID of the User who activated this order. This is a relationship field. **Relationship Name** ActivatedBy **Relationship Type** Lookup **Refers To** User
`ActivatedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort, Update **Description** Date and time when the order was activated.
`BillingAddress`	**Type** address

Field Name	Details
	Properties Filter, Nillable **Description**
`BillingCity`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** City for the billing address for this order. Maximum size is 40 characters.
`BillingCountry`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Country for the billing address for this order. Maximum size is 80 characters.
`BillingCountryCode`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** ISO country code for the billing address for this order.
`BillingEmailAddress`	**Type** email **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Email address for this order's billing address. To access Commerce Orders fields, your org must have a Salesforce Order Management license or a B2B Commerce License. This field is available in API v48.0 and later.
`BillingGeocodeAccuracy`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Accuracy level of the geocode of the address.

Field Name	Details
`BillingLatitude`	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** Used with `BillingLongitude` to specify the precise geolocation of a billing address. Acceptable values are numbers between –90 and 90 with up to 15 decimal places.
`BillingLongitude`	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** Used with `BillingLatitude` to specify the precise geolocation of a billing address. Acceptable values are numbers between –180 and 180 with up to 15 decimal places.
`BillingPhoneNumber`	**Type** phone **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Phone number for this order's billing address. To access Commerce Orders fields, your org must have a Salesforce Order Management license or a B2B Commerce License. This field is available in API v48.0 and later.
`BillingPostalCode`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Postal code for the billing address for this order. Maximum size is 20 characters.
`BillingState`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** State for the billing address for this order. Maximum size is 80 characters.

Field Name	Details
`BillingStateCode`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** ISO state code for the order's billing address.
`BillingStreet`	**Type** textarea **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Street address for the billing address.
`BillToContactId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** ID of the contact that the order is billed to.
`CompanyAuthorizedById`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** ID of the user who authorized the account associated with the order. This is a relationship field. **Relationship Name** CompanyAuthorizedBy **Relationship Type** Lookup **Refers To** User
`CompanyAuthorizedDate`	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The date on which your organization authorized the order.

Field Name	Details
`ContractId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** ID of the contract associated with this order. Can only be updated when the order's `StatusCode` value is *`Draft`*. This is a relationship field. **Relationship Name** Contract **Relationship Type** Lookup **Refers To** Contract
`CurrencyIsoCode`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update **Description** Three-letter ISO 4217 currency code.
`CustomerAuthorizedById`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** ID of the contact who authorized the order. This is a relationship field. **Relationship Name** CustomerAuthorizedBy **Relationship Type** Lookup **Refers To** Contact
`CustomerAuthorizedDate`	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update

Field Name	Details
	Description Date on which the contact authorized the order.
`Description`	**Type** textarea **Properties** Create, Nillable, Update **Description** Description of the order.
`EffectiveDate`	**Type** date **Properties** Create, Filter, Group, Sort, Update **Description** Date at which the order becomes effective. Label is **Order Start Date**.
`EndDate`	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Date at which the order ends. Label is **Order End Date**.
`GrandTotalAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Sum of `TotalAmount` and `TotalTaxAmount`. To access Commerce Orders fields, your org must have a Salesforce Order Management license or a B2B Commerce License. This field is available in API v48.0 and later.
`IsReductionOrder`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort **Description** Read-only. Determines whether an order is a reduction order. Label is **Reduction Order**.

Field Name	Details
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last interacted with this record, directly or indirectly. Some sample scenarios are:
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed this record or list view. If this value is null, it's possible that the user only accessed this record or list view (`LastReferencedDate`), but not viewed it.
`Name`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Name for this order.
`OpportunityId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** ID for the opportunity that's associated with this order.
`OrderedDate`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The date and time that the order was placed. To access Commerce Orders fields, your org must have a Salesforce Order Management license or a B2B Commerce License. This field is available in API v48.0 and later.

Field Name	Details
`OrderManagementReferenceIdentifier`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** This field is used by Order Management to store the external reference Identifier for B2C Commerce orders. On creation, the B2C Integration sets this value to *`B2C realm ID`* + "_" + *`B2C instance ID`* + "@" + *`B2C Commerce catalog/domain ID`* + "@" + *`B2C Commerce order number`*. Otherwise, it isn't set. When you create an OrderSummary, if you don't specify an ExternalReferenceIdentifier value, it's set to this value. If this value is null, then the system generates a value for ExternalReferenceIdentifier. This value isn't required to be unique in an organization, but the OrderSummary ExternalReferenceIdentifier is. This field is available in API version 56.0 and later.
`OrderNumber`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Order number assigned to this order (not the unique, system-generated ID assigned during creation). Maximum size is 30 characters.
`OrderReferenceNumber`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Order reference number assigned to this order. Maximum size is 80 characters.
`OriginalOrderId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Nillable, Sort **Description** Optional. ID of the original order that a reduction order is reducing, if the reduction order is reducing a single order. Label is **Original Order**. Editable only if `isReductionOrder` is *`true`*. If the reduction order is reducing more than one order, leave blank. This is a relationship field.

Field Name	Details
	Relationship Name OriginalOrder **Relationship Type** Lookup **Refers To** Order
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. ID of the User or queue that owns this order. This is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User
`PaymentTermId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Nillable, Sort **Description** The ID of the related payment term. This field is available in API version 55.0 and later. This field is available if Subscription Management is enabled in your org. This is a relationship field. **Relationship Name** PaymentTerm **Relationship Type** Lookup **Refers To** PaymentTerm
`PoDate`	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update

Field Name	Details
	Description Date of the purchase order.
`PoNumber`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Number identifying the purchase order. Maximum is 80.
`Pricebook2Id`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Nillable, Sort, Update **Description** Required. ID of the price book associated with this order. This is a relationship field. **Relationship Name** Pricebook2 **Relationship Type** Lookup **Refers To** Pricebook2
`QuoteId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** ID of the quote that's associated with this order. If you set `QuoteId` to null, `QuoteLineItemId` on all of the order's child order products is set to null.
`RecordTypeId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** ID of the record type assigned to this order.

Field Name	Details
RelatedOrderId	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The original order that a change order was created from. To access Commerce Orders fields, your org must have a Salesforce Order Management license or a B2B Commerce License. This field is available in API v48.0 and later. This field is a relationship field. **Relationship Name** RelatedOrder **Relationship Type** Lookup **Refers To** Order
SalesChannelId	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Reference to a sales channel entity. To access Commerce Orders fields, your org must have a Salesforce Order Management license or a B2B Commerce License. This field is available in API v48.0 and later.
SalesStoreId	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** ID of the RetailStore or WebStore associated with this Order. This field is a polymorphic relationship field. To access Commerce Orders fields, your org must have a Salesforce Order Management license or a B2B Commerce License. This field is available in API v46.0 and later. **Relationship Name** SalesStore

Field Name	Details
	Relationship Type Lookup **Refers To** WebStore
`ShippingAddress`	**Type** address **Properties** Filter, Nillable **Description** Shipping address for the order.
`ShippingCity`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** City of the shipping address. Maximum size is 40 characters.
`ShippingCountry`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Country of the shipping address. Maximum size is 80 characters.
`ShippingCountryCode`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** ISO country code for the order's shipping address.
`ShippingGeocodeAccuracy`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Accuracy level of the geocode of the shipping address.
`ShippingLatitude`	**Type** double

Field Name	Details
	Properties Create, Filter, Nillable, Sort, Update **Description** Used with `ShippingLongitude` to specify the precise geolocation of a shipping address. Acceptable values are numbers between –90 and 90 with up to 15 decimal places.
`ShippingLongitude`	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** Used with `ShippingLatitude` to specify the precise geolocation of an address. Acceptable values are numbers between –180 and 180 with up to 15 decimal places.
`ShippingPostalCode`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Postal code of the shipping address. Maximum size is 20 characters.
`ShippingState`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** State of the shipping address. Maximum size is 80 characters.
`ShippingStateCode`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** ISO state code for the order's shipping address.
`ShippingStreet`	**Type** textarea **Properties** Create, Filter, Group, Nillable, Sort, Update

Field Name	Details
	Description Street address of the shipping address. Maximum of 255 characters.
`ShipToContactId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** ID of the contact that the order is shipped to.
`Status`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Picklist of values that indicate order status. Each value is within one of two status categories defined in `StatusCode`. For example, the status picklist might contain *`Draft`*, *`Ready for Review`*, and *`Ready for Activation`* values with a `StatusCode` of *`Draft`*.
`StatusCode`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort, Update **Description** The status category for the order. An order can be either *`Draft`* or *`Activated`*. Label is **Status Category**.

`TaxLocaleType`	**Type**
	picklist
	Properties
	Create, Filter, Group, Nillable, Restricted picklist, Sort, Update
	Description
	The type of tax calculation that Salesforce uses for the order's order items. VAT regions use gross tax, which includes tax in all sale amounts. US regions use net tax, which calculates tax separately from the initial sale amount and then adds the sale and tax amounts together in a total.
	Use `TaxLocaleType` to determine which types of tax fields to show on your order. If `TaxLocaleType` is null, the order shows all tax fields.
	Gross Tax Fields
	`TotalAdjDeliveryAmtWithTax`
	`TotalAdjProductAmtWithTax`

Field Name	Details

	`TotalProductAdjDistAmtWithTax`
	`TotalDeliveryAdjDistAmtWithTax`
	To access Commerce Orders fields, your org must have a Salesforce Order Management license or a B2B Commerce License.
	This field is available in API v49.0 and later.

`TotalAdjDeliveryAmtWithTax`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Sum of delivery line amounts, delivery line adjustments, and tax. Order products with null Type fields aren't included. This is a gross tax field. To access Commerce Orders fields, your org must have a Salesforce Order Management license or a B2B Commerce License. This field is available in API v49.0 and later.
`TotalAdjProductAmtWithTax`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Sum of product line amounts, line adjustments, and tax. Order products with null Type fields aren't included. This is a gross tax field. To access Commerce Orders fields, your org must have a Salesforce Order Management license or a B2B Commerce License. This field is available in API v49.0 and later.
`TotalAdjustedDeliveryAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Sum of delivery line amounts and delivery line adjustments. Order products with null Type fields aren't included. To access Commerce Orders fields, your org must have a Salesforce Order Management license or a B2B Commerce License. This field is available in API v48.0 and later.

Field Name	Details
`TotalAdjustedDeliveryTaxAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Sum of delivery line tax amounts and delivery line tax adjustments. To access Commerce Orders fields, your org must have a Salesforce Order Management license or a B2B Commerce License. This field is available in API v48.0 and later.
`TotalAdjustedProductAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Sum of product line amounts and line adjustments. Order products with null Type fields aren't included. To access Commerce Orders fields, your org must have a Salesforce Order Management license or a B2B Commerce License. This field is available in API v48.0 and later.
`TotalAdjustedProductTaxAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Sum of line tax amounts and line tax adjustments. Order products with null Type fields aren't included. To access Commerce Orders fields, your org must have a Salesforce Order Management license or a B2B Commerce License. This field is available in API v48.0 and later.
`TotalAmount`	**Type** currency **Properties** Filter, Sort **Description** The net total amount for the order products associated with this order. This field is available in API v48.0 and later.

Field Name	Details
`TotalDeliveryAdjDistAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Roll-up of the order's delivery adjustment distributed amounts. Used only when the Order Adjustment Group has a Type value of Header. To access Commerce Orders fields, your org must have a Salesforce Order Management license or a B2B Commerce License. This field is available in API v48.0 and later.
`TotalDeliveryAdjDistAmtWithTax`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Roll-up of the order's delivery adjustment distributed amounts and tax. Used only when the Order Adjustment Group has a Type value of Header. This is a gross tax field. To access Commerce Orders fields, your org must have a Salesforce Order Management license or a B2B Commerce License. This field is available in API v49.0 and later.
`TotalDeliveryAdjDistTaxAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Roll-up of the order's delivery adjustment distributed tax amounts. Used only when the Order Adjustment Group has a Type value of Header. To access Commerce Orders fields, your org must have a Salesforce Order Management license or a B2B Commerce License. This field is available in API v48.0 and later.
`TotalProductAdjDistAmount`	**Type** currency **Properties** Filter, Nillable, Sort

Field Name	Details
	Description Roll-up of the order's product adjustment distributed amounts. Order products with null Type fields aren't included. Used only when the Order Adjustment Group has a Type value of Header. To access Commerce Orders fields, your org must have a Salesforce Order Management license or a B2B Commerce License. This field is available in API v48.0 and later.
`TotalProductAdjDistAmtWithTax`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Roll-up of the order's product adjustment distributed amounts. Order products with null Type fields aren't included. Used only when the Order Adjustment Group has a Type value of Header. This is a gross tax field. To access Commerce Orders fields, your org must have a Salesforce Order Management license or a B2B Commerce License. This field is available in API v49.0 and later.
`TotalProductAdjDistTaxAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Roll-up of the order's product adjustment distributed tax amounts. Order products with null Type fields aren't included. Used only when the Order Adjustment Group has a Type value of Header. To access Commerce Orders fields, your org must have a Salesforce Order Management license or a B2B Commerce License. This field is available in API v48.0 and later.
`TotalTaxAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** Roll up of all tax on the order. Includes delivery taxes, price adjustment taxes, and product taxes.

Field Name	Details
	To access Commerce Orders fields, your org must have a Salesforce Order Management license or a B2B Commerce License. This field is available in API v48.0 and later.
`Type`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** If you want to show more information about your order, you can add custom values to the `Type` picklist. By default, the `Type` field doesn't perform any actions or show any values.

Usage

The `Status` field specifies the current state of an order. Status strings represent its current state (*`Draft`* or *`Activated`*).

When a client application creates an order, the `Status Code` must be *`Draft`* and the `Status` must be any value that corresponds to a `Status Code` of *`Draft`*. The application can then activate an order by updating it and setting the value in its `Status` field to an *`Activated`* state; however, the `Status` field is the only field you can update when activating the order.

After an order is activated, your client application can change the `Status` back to the *`Draft`* state—but only if the order doesn't have any child reduction order products. Your client application can delete orders when the `Status` is *`Draft`* but not when its `Status` is *`Activated`*.

Client applications can use the API to create, update, delete, and query any Attachment associated with an order.

Orders Without Price Books

If your organization manages products and Price books in an external platform, you can use Salesforce API to create orders and order items without values for their Price book and Price book entry fields. This feature is available only for Salesforce orgs with the B2B Commerce, B2B Commerce Starter, B2B Commerce Growth, or B2B Commerce Plus packages. Admins enable orders without Price books by going to Salesforce Order Settings and selecting the Optional Price Book setting.

In a standard order, Salesforce prompts the sales rep to select a Price book when they add the first order product to the order. The sales rep can then add order products that have price book entries in the selected price book. In an order without a Price book, Salesforce hides the order's Add Products button and Edit Products button so that sales reps must manage their products and price books using their external system.

You can create orders without Price books only by creating an order with Salesforce API and leaving the `Pricebook2Id` field null. Orders without Price books follow several different guidelines compared to standard orders.

- Orders without price books don't support reduction orders or change orders.
- Order products without price book entries require list prices.
- Orders without price books support only order items without price book entries. Orders with price books support only order items with price book entries.
- **Important:** Orders without Price books are supported with B2B licenses only. Salesforce Order Management requires price books for orders and price book entries for order products.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

OrderChangeEvent (API version 44.0)
Change events are available for the object.

OrderFeed (API version 29.0)
Feed tracking is available for the object.

OrderHistory
History is available for tracked fields of the object.

OrderOwnerSharingRule
Sharing rules are available for the object.

OrderAction

Indicates the type of order, such as a new sale or a cancellation. This object is available in API version 55.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

This object is available if Subscription Management is enabled in your org.

Fields

Field	Details
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name given to the order action.
`OffsetOrderItemId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ID of the previous order item that is being modified by the business action. For example, the order that is being canceled. This is a relationship field.

Field	Details
	Relationship Name OffsetOrderItem **Relationship Type** Lookup **Refers To** OrderItem
`OrderId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The order containing the order item that implements the business action. This is a relationship field. **Relationship Name** Order **Relationship Type** Lookup **Refers To** Order
`SourceAssetId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The asset that is changed as a result of the business action. For example, the asset that is being canceled. This is a relationship field. **Relationship Name** SourceAsset **Relationship Type** Lookup **Refers To** Asset
`Type`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update

Field	Details
	Description The business action that created the order product. Possible values are: • `Cancellation` • `New Sale` • `No Change`—A child product was added to the bundle, but the top-level product in the bundle was otherwise unchanged. • `Quantity Amendment` • `Renewal`

ProductDecompEnrichmentRule

Represents mappings between fields and attributes. Enrichment rules are part of a decomposition rule, and are used to propagate data to fulfillment order lines. This object is available in API version 61.0 and later.

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.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`CalculationDefinitionId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** An expression set or a decision matrix that calculates the destination value. This field is a polymorphic relationship field. **Relationship Name** CalculationDefinition **Refers To** DecisionMatrixDefinition, ExpressionSet
`CalculationMethod`	**Type** picklist

Field	Details
	Properties Create, Filter, Group, Restricted picklist, Sort, Update **Description** The type of enrichment rule. Valid values are: • `Ad-verbatim`—As Is • `Static-Lookup`—List Lookup
`DecompositionRuleId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The identifier of the decomposition rule. This field is a relationship field. **Relationship Name** DecompositionRule **Relationship Type** Parent-child **Refers To** ProductFulfillmentDecompRule (the master object)
`DestinationApiName`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The API name of the destination field or attribute code of an attribute.
`DestinationAttributeDefinitionId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** For internal use only. This field is a relationship field. **Relationship Name** DestinationAttributeDefinition **Refers To** AttributeDefinition

Field	Details
`DestinationContextTag`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The name of the destination context definition.
`DestinationType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The destination type for mapping. Valid values are: • `Attribute` • `Field`
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The most recent date when a user referenced this record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The most recent date when a user viewed this record
`ListMappingGroupId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** For internal use only. This field is a relationship field. **Relationship Name** ListMappingGroup

Field	Details
	Refers To ValTfrmGrp
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name of the decomposition rule.
`RuleEnforcement`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Specifies whether the rule applies to all fulfillment requests or only to specific ones. Valid values are: • `AllFulfillmentRequests` • `InitialFulfillmentRequest` This field is available in API version 63.0 and later.
`SourceApiName`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The API name of the source field or attribute code of an attribute.
`SourceAttributeDefinitionId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** For internal use only. This field is a relationship field. **Relationship Name** SourceAttributeDefinition **Refers To** AttributeDefinition

Field	Details
SourceContextTag	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The source type for the context definition.
SourceType	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The source type for mapping. Valid values are: • Attribute • Field

ProductFulfillmentDecompRule

Represents a rule that determines how an order is broken into sub-orders with specific technical details that help in order fulfillment. It can be applied to a commercial or a technical product. This object is available in API version 61.0 and later.

Supported Calls

create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), undelete(), update(), upsert()

Fields

Field	Details
DestinationProductId	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The destination product for the decomposition rule. This field is a relationship field. **Relationship Name** DestinationProduct

Field	Details
	Relationship Type Lookup **Refers To** Product2
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The most recent date when a user referenced this record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The most recent date when a user viewed this record.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** The name of the decomposition rule.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The ID of the user who created the record. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User

Field	Details
`Priority`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The priority of the decomposition rule. Decomposition rules are executed in order of priority.
`SourceProductClassificationId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The classification of the source product that's used for decomposition. This field is available in API version 62.0 and later. This field is a relationship field. **Relationship Name** SourceProductClassification **Refers To** ProductClassification
`SourceProductId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** Source product for the decomposition rule. This field is a relationship field. **Relationship Name** SourceProduct **Relationship Type** Lookup **Refers To** Product2

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ProductFulfillmentDecompRuleShare **on page 1684**

Sharing is available for the object.

ProductFulfillmentScenario

Represents a link between a product and the corresponding group of fulfillment steps that's necessary to fulfill that product. This object is available in API version 61.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`Action`	**Type** multipicklist **Properties** Create, Filter, Nillable, Update **Description** For internal use only. Valid values are: • `Add` • `Amend` • `Cancel` • `NoChange` • `Renew`
`FulfillmentStepDefnGroupId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The fulfillment step definition group associated with the product fulfillment scenario. This field is a relationship field. **Relationship Name** FulfillmentStepDefnGroup **Relationship Type** Lookup **Refers To** FulfillmentStepDefinitionGroup
`LastReferencedDate`	**Type** dateTime

Field	Details
	Properties Filter, Nillable, Sort **Description** For internal use only.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** For internal use only.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** The name of the product fulfillment scenario.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** For internal use only. This field is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User
`ProductId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** The product associated with the product fulfillment scenario. This field is a relationship field.

Field	Details
	Relationship Name Product **Relationship Type** Lookup **Refers To** Product2

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ProductFulfillmentScenarioShare **on page 1684**
Sharing is available for the object.

SalesTransactionFulfillReq

Represents the statuses of all the sub-orders that belong to the selected commercial or technical product. This object is available in API version 62.0 and later.

Supported Calls

`describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Fields

Field	Details
`AssetizationStatus`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Specifies the status of the assetization. Valid values are: • `Completed` • `Failed` • `InProgress` • `NotStarted` • `Rejected`

Field	Details
DecompositionStatus	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Specifies the status of the decomposition. Valid values are: • Completed • Failed • InProgress • NotStarted • Rejected
Name	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name of the sales transaction fulfillment request.
OrchestrationGroupKey	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The identifier of the group of sales transactions that require synchronization before processing. This field is available in API version 63.0 and later.
OwnerId	**Type** reference **Properties** Filter, Group, Sort **Description** The ID of the user who created the request record. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User

Field	Details
`PlanCompositionStatus`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** For internal use only. Valid values are: • `Completed` • `Failed` • `InProgress` • `NotStarted` • `Rejected`
`PlanExecutionStatus`	**Type** picklist **Properties** Filter, Group, Nillable, Sort **Description** Specifies the status of the plan execution. Valid values are: • `InProgress` • `Frozen` • `Freezing`
`PlanId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The identifier of the plan. This field is a relationship field. **Relationship Name** Plan **Refers To** FulfillmentPlan
`PreviousRequestId`	**Type** reference **Properties** Filter, Group, Nillable, Sort

Field	Details
	Description The identifier of the previous fulfillment request. This field is a relationship field. **Relationship Name** PreviousRequest **Refers To** SalesTransactionFulfillReq
`ReferenceObjectId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The object associated with the sales transaction fulfillment request. This field is a polymorphic relationship field. **Relationship Name** ReferenceObject **Refers To** Order, Quote
`SalesTransactionType`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Specifies the type of sales transaction that's processed by the fulfillment request. Valid values are: • `Revenue Cloud Order` • `CME Managed Package Order` • `Commerce Order`
`Status`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Specifies the overall status of the fulfillment. Valid values are: • `Created` • `Freezing`

Field	Details
	• `Frozen` • `Fulfilled` • `Fulfilling` • `Rejected` • `Superseded`

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

SalesTransactionFulfillReqShare on page 1684
Sharing is available for the object.

ValTfrm

Represents mappings between fields and attributes. Enrichment rules are part of a decomposition rule, and are used to propagate data to fulfillment order lines. This object is available in API version 61.0 and later.

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.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`InputDate`	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The date of value entry.
`InputDatetime`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update

Field	Details
	Description The time of value entry.
`InputNumber`	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The value of input number.
`InputPicklistValueId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The identifier of the input list of values. This field is a relationship field. **Relationship Name** InputPicklistValue **Refers To** AttributePicklistValue
`InputString`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The value of input text.
`IsInputBoolean`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates if a value was entered (`true`) or not (`false`). The default value is `false`.
`IsOutputBoolean`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update

Field	Details
	Description Indicates if there was an output value (`true`) or not (`false`). The default value is `false`.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** The name of the related transformation group.
`OutputDate`	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The date of matched output.
`OutputDatetime`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** The time of matched output.
`OutputNumber`	**Type** double **Properties** Create, Filter, Nillable, Sort, Update **Description** The value of output number.
`OutputPicklistValueId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The identifier of the output list of values. This field is a relationship field. **Relationship Name** OutputPicklistValue

Field	Details
	Refers To AttributePicklistValue
`OutputString`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The value of output text.
`ValueTransformGroupId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The identifier of the related transformation group. This field is a relationship field. **Relationship Name** ValueTransformGroup **Relationship Type** Master-detail **Refers To** ValTfrmGrp (the master object)

ValTfrmGrp

Represents a rule that determines how an order is broken into sub-orders with specific technical details that help in order fulfillment. The rule can be applied to a commercial or a technical product. This object is available in API version 61.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
`DestinationPrimitiveType`	**Type** picklist

Field	Details
	Properties Create, Filter, Group, Restricted picklist, Sort, Update **Description** The data type of the output value. Valid values are: • `Boolean` • `Currency` • `Date` • `Datetime` • `Number` • `Percent`—Picklist Value • `Picklist` • `Text`
`IsDestinationEnumerated`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the output is a list of values (`true`) or not (`false`). The default value is `false`.
`IsSourceEnumerated`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the input is from a list of values (`true`) or not (`false`). The default value is `false`.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The most recent date when a user referenced this record.
`LastViewedDate`	**Type** dateTime

Field	Details
	Properties Filter, Nillable, Sort **Description** The most recent date when a user viewed this record.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** The name of the list mapping.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The ID of the user who owns this record. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
`SourcePrimitiveType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The data type of the input value. Valid values are: • `Boolean` • `Currency` • `Date` • `Datetime` • `Number` • `Percent`—Picklist Value • `Picklist` • `Text`

Field	Details
`UsageType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The data mapping feature that uses this value transformation group. Valid value is `DFOListMapping`.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

ValTfrmGrpShare on page 1684
Sharing is available for the object.

Dynamic Revenue Orchestrator Standard Invocable Actions

Use standard invocable actions to submit an order or a sales transaction to Dynamic Revenue Orchestrator (DRO) for fulfillment.

Submit Order Action
Submit an order to Dynamic Revenue Orchestrator (DRO) for fulfillment.

Submit Sales Transaction Action
Initiate the fulfillment process of any sales transaction such as a quote, an order, or an order summary.

SEE ALSO:

Actions Developer Guide: Overview

REST API Developer Guide: Invocable Actions Standard

Submit Order Action

Submit an order to Dynamic Revenue Orchestrator (DRO) for fulfillment.

By using the Submit Order action, you can perform:

- Order decomposition
- Fulfillment orchestration that's driven through message queues
- Dynamic plan composition that's based on the incoming order

This action is available in API version 61.0 and later.

Special Access Rules

The Submit Order action is available in Enterprise, Unlimited, and Developer Editions of Revenue Cloud. See the required permissions to access and call this invocable action.

Supported REST HTTP Methods

URI

`/services/data/v63.0/actions/standard/submitOrder`

Formats

JSON, XML

HTTP Methods

POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`orderId`	**Type** string **Description** Required. ID of the order to submit to DRO.
`callType`	**Type** string **Description** Optional. Mode that the order intake must be processed in. Valid values are: • `Synchronous` • `Asynchronous` The default value is `Asynchronous`.
`contextId`	**Type** string **Description** Optional. ID of the hydrated context. See Context Service.

Outputs

Output	Details
`errorCode`	**Type** string **Description** Error code for the failed request, if any.
`fulfillmentPlanId`	**Type** string **Description** ID of the orchestrated fulfillment plan that's generated. Returned only if the `callType` value is `Synchronous`.
`requestId`	**Type** string **Description** Unique ID of the invocation request that helps identify a single request.
`submitStatus`	**Type** string **Description** Submit status of the invocation request. Valid values are: • `SUCCESS` • `ERROR` • `SUBMITTED` • `REJECTED`
`usedContextId`	**Type** string **Description** Hydrated context ID that's used in this request, which can be different from the `contextId` input.

Example

POST

This example shows a sample request.

```
{
    "inputs": [
        {
```

```
            "orderId": "801RM0000007yGaYAI",
            "callType": "Synchronous"
        }
    ]
}
```

This example shows a sample response when the call type is synchronous.

```
[
    {
   "actionName":"submitOrder"
   "errors":NULL
   "invocationId":NULL
   "isSuccess":true
   "outputValues":{
   "requestId":"a161cfda-868c-41d2-b589-7c7d7ff2d4c1"
   "submitStatus":"SUCCESS"
   "usedContextId":"e275e930923106ee7e39cbfa232e38252bd4d63f4ea2dd956b7301e243554134"
   "fulfillmentPlanId":"13VZM00000000062AA"
}
]
```

This example shows a sample response when the call type is asynchronous or isn't specified.

```
[
{
    "actionName": "submitOrder",
    "errors": null,
    "isSuccess": true,
    "outputValues": {
    "submitStatus": "SUBMITTED"
    "requestId": "a161cfda-868c-41d2-b589-7c7d7ff2d4c1"
}
}
]
```

This example shows a sample response when a validation error occurs.

```
[
    {
    "actionName": "submitOrder",
    "errors": [
    {
        "statusCode": "UNKNOWN_EXCEPTION",
        "message": "Missing required input parameter: orderId",
        "fields": []
     }
  ],
        "invocationId": null,
        "isSuccess": false,
        "outputValues": {
          "requestId": "4c7d8ebb-6b0b-4852-a8a0-b67e0d36a73e",
          "errorCode": "DRO_INTERNAL_ERROR"
        },
    }
]
```

Explainability Action Logs

To troubleshoot or debug errors, retrieve a list of explainability action logs. See Action Logs.

Get logs for order intake

This resource example includes sample query parameters to retrieve the action logs for an order intake.

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/decision-explainer/action-logs?applicationSubType=DroSubmit&applicationType=7&processType=DroSubmit&primaryFilter=801xx000003XAG
```

This example shows the sample response. The `actionLog` property contains the action logs.

```
{
  "actionLogs": [
    {
      "actionContextCode": "801NA000000XKPUYA4",
      "actionLog": {
        "OrderIntakeStatus": "",
        "OrderIntakeStatusMessage": "",
        "OrderId": "801NA000000XKPUYA4",
        "SubmitMode": "Synchronous",
        "UniqueRequestId": "d00c2aa7-56b0-411a-9b51-83d9fe2e4440",
       "ContexId": "a0ca3c2296c82ad071a5efa83e8974793910f44ebddb21d37f007ce4889b65d7",

        "DesActionSpecDevName": "DroSubmitAction",
        "ContextDefinition": "SalesTransactionContext__stdctx"
      },
      "additionalFilter": "undef",
      "applicationLogDate": "Thu Mar 28 12:36:26 GMT 2024",
      "applicationSubtype": "DroSubmit",
      "applicationType": "7",
      "explainabilitySpecName": "DroSubmitAction",
      "isChunked": false,
      "name": "DroSubmitAction",
      "primaryFilter": "801NA000000XKPUYA4",
      "processType": "DroSubmit",
      "secondaryFilter": "undef",
      "uniqueIdentifier": "02c4bae9-d8b0-42f6-b031-e983d4247c76"
    }
  ],
  "queryMore": ""
}
```

Get logs for decomposition scope

This resource example includes sample query parameters to retrieve the decomposition-related action logs.

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/decision-explainer/action-logs?applicationSubType=DroDecomp&applicationType=7&processType=DroDecomp&primaryFilter=801xx000003XAG
```

This example shows the sample response. The `actionLog` property contains the action logs.

```
{
  "actionLogs": [
    {
      "actionContextCode": "801NA000000XKPeYAO",
      "actionLog": {
        "CandidateDecompositionRules": [
```

```
        {
          "decompRuleId": "13UNA0000004C932AE",
          "SourceProductId": "01tNA0000007NP5YAM",
          "AssociatedOlis": [
            "802NA000002lWjOYAU"
          ],
          "DestinationProductId": "01tNA0000007NOvYAM",
          "DestinationProductScope": "Order",
          "ConfiguredContextRuleId": "null"
        },
        {
          "decompRuleId": "13UNA0000004ITY2A2",
          "SourceProductId": "01tNA0000007NP5YAM",
          "AssociatedOlis": [
            "802NA000002lWjOYAU"
          ],
          "DestinationProductId": "01tNA0000007NOqYAM",
          "DestinationProductScope": "Order",
          "ConfiguredContextRuleId": "null"
        },
        {
          "decompRuleId": "13UNA0000004C942AE",
          "SourceProductId": "01tNA0000007NPAYA2",
          "AssociatedOlis": [
            "802NA000002lWjKYAU"
          ],
          "DestinationProductId": "01tNA0000007NOwYAM",
          "DestinationProductScope": "Order",
          "ConfiguredContextRuleId": "null"
        },
        {
          "decompRuleId": "13UNA0000004ITd2AM",
          "SourceProductId": "01tNA0000007NPAYA2",
          "AssociatedOlis": [
            "802NA000002lWjKYAU"
          ],
          "DestinationProductId": "01tNA0000007NP0YAM",
          "DestinationProductScope": "Order",
          "ConfiguredContextRuleId": "null"
        },
        {
          "decompRuleId": "13UNA0000004C952AE",
          "SourceProductId": "01tNA0000007NPBYA2",
          "AssociatedOlis": [
            "802NA000002lWjLYAU"
          ],
          "DestinationProductId": "01tNA0000007NOxYAM",
          "DestinationProductScope": "Order",
          "ConfiguredContextRuleId": "9QwNA0000000CTu0AM"
        },
        {
          "decompRuleId": "13UNA0000004C982AE",
          "SourceProductId": "01tNA0000007NPCYA2",
          "AssociatedOlis": [
```

```
            "802NA000002lWjNYAU"
          ],
          "DestinationProductId": "01tNA0000007NOqYAM",
          "DestinationProductScope": "Order",
          "ConfiguredContextRuleId": "null"
        },
        {
          "decompRuleId": "13UNA0000004C992AE",
          "SourceProductId": "01tNA0000007NPCYA2",
          "AssociatedOlis": [
            "802NA000002lWjNYAU"
          ],
          "DestinationProductId": "01tNA0000007NOrYAM",
          "DestinationProductScope": "Order",
          "ConfiguredContextRuleId": "9QwNA0000000CTt0AM"
        }
      ],
      "SelectedDecompositionRules": [
        {
          "OliId": "802NA000002lWjKYAU",
          "DecompositionRuleIds": [
            "13UNA0000004C942AE",
            "13UNA0000004ITd2AM"
          ]
        },
        {
          "OliId": "802NA000002lWjOYAU",
          "DecompositionRuleIds": [
            "13UNA0000004C932AE",
            "13UNA0000004ITY2A2"
          ]
        },
        {
          "OliId": "802NA000002lWjNYAU",
          "DecompositionRuleIds": [
            "13UNA0000004C982AE"
          ]
        }
      ],
      "OliScopeDetails": [
        {
          "OliId": "802NA000002lWjLYAU",
          "ParentOliId": "802NA000002lWjOYAU",
          "BundleRootOli": "802NA000002lWjOYAU"
        },
        {
          "OliId": "802NA000002lWjMYAU",
          "ParentOliId": "802NA000002lWjOYAU",
          "BundleRootOli": "802NA000002lWjOYAU"
        },
        {
          "OliId": "802NA000002lWjKYAU",
          "ParentOliId": "802NA000002lWjOYAU",
          "BundleRootOli": "802NA000002lWjOYAU"
```

```
      },
      {
        "OliId": "802NA000002lWjNYAU",
        "ParentOliId": "802NA000002lWjOYAU",
        "BundleRootOli": "802NA000002lWjOYAU"
      }
    ],
    "FoliComputationDetails": [
      {
        "FoliId": "0a4NA000003HYbWYAW",
        "ComputedAction": "AMEND",
        "ComputedQuantity": "3.0"
      },
      {
        "FoliId": "0a4NA000003HYbXYAW",
        "ComputedAction": "AMEND",
        "ComputedQuantity": "6.0"
      },
      {
        "FoliId": "0a4NA000003HYbYYAW",
        "ComputedAction": "AMEND",
        "ComputedQuantity": "3.0"
      },
      {
        "FoliId": "0a4NA000003HYbZYAW",
        "ComputedAction": "AMEND",
        "ComputedQuantity": "3.0"
      }
    ],
    "FlsrDetails": [
      {
        "FlsrId": "16ANA000005idR82AI",
        "FlsrGuid": "WQoYClnAszG9axKxWxoe",
        "SourceId": "802NA000002lWjOYAU",
        "FoliId": "0a4NA000003HYbWYAW"
      },
      {
        "FlsrId": "16ANA000005idR92AI",
        "FlsrGuid": "80xWFpBK8sDmTFRV3Dn0",
        "SourceId": "802NA000002lWjOYAU",
        "FoliId": "0a4NA000003HYbXYAW"
      },
      {
        "FlsrId": "16ANA000005idRA2AY",
        "FlsrGuid": "nAql9VD3KiziZTFrfYXq",
        "SourceId": "802NA000002lWjNYAU",
        "FoliId": "0a4NA000003HYbXYAW"
      },
      {
        "FlsrId": "16ANA000005idRB2AY",
        "FlsrGuid": "wgCJT6NJ3ESolFXP5ajQ",
        "SourceId": "802NA000002lWjKYAU",
        "FoliId": "0a4NA000003HYbYYAW"
      },
```

```
          {
            "FlsrId": "16ANA000005idRC2AY",
            "FlsrGuid": "bvVhA34WvTBOYbG4q2Fj",
            "SourceId": "802NA000002lWjKYAU",
            "FoliId": "0a4NA000003HYbZYAW"
          }
        ],
        "OrderId": "801NA000000XKPeYAO",
        "SubmitMode": "Synchronous",
        "UniqueRequestId": "b9ac5971-0d71-45c6-a18e-f472976eaeae",
       "ContextId": "1422049ffa3cc42d8c060b9a7732be360bd62a346652cda9ab6e5fd54cd03eca",

        "DesActionSpecDevName": "DroDecompScopeAction",
        "ContextDefinition": "SalesTransactionContext__stdctx"
      },
      "additionalFilter": "undef",
      "applicationLogDate": "Fri Mar 29 10:02:26 GMT 2024",
      "applicationSubtype": "DroDcmp",
      "applicationType": "7",
      "explainabilitySpecName": "DroDecompScopeAction",
      "isChunked": false,
      "name": "DroDecompScopeAction",
      "primaryFilter": "801NA000000XKPeYAO",
      "processType": "DcmpScp",
      "secondaryFilter": "undef",
      "uniqueIdentifier": "c1e11bc0-53e5-4f62-9ae8-1096e79a62b6"
    }
  ],
  "queryMore": ""
}
```


Get logs for decomposition enrichment

This resource example includes sample query parameters to retrieve the logs for decomposition enrichment tasks. For example, conversion of order items to fulfillment order line items.

```
https://yourInstance.salesforce.com/services/data/v60.0/connect/dro/explainability/actionLog?applicationSubtype=DroDcmp&applicationType=7&processType=DcmpScp&primaryFilter=801NA000000XKPeYAO&secondaryFilter=0a4NA000003HYbZYAW
```

The `secondaryFilter` property is optional. If this property is specified, the API returns the enrichment rule details for a decomposition rule. If this property is unspecified, the API returns all enrichment rule details for the order.

This example shows the sample response. The `actionLog` property contains the action logs.


```
{
  "actionLogs": [
    {
      "actionContextCode": "801NA000000XKPUYA4",
      "actionLog": {
        "FlsrUid": "1FxfIhMq53OIpE9Hzs5w",
        "EnrichmentRulesExecutionDetails": [
          {
            "EnrichmentRuleId": "13TNA00000000eG2AQ",
            "DecompRuleId": "13UNA0000004C942AE",
            "FlsrUid": "1FxfIhMq53OIpE9Hzs5w",
            "SourceType": "Attribute",
```

```
      "SourceAttributeId": "0tjNA0000004CygYAE",
      "TargetType": "Attribute",
      "TargetAttributeId": "0tjNA0000004CyDYAU",
      "CalculationMethod": "Ad-verbatim",
      "ExecutionStatus": "SUCCESS"
    },
    {
      "EnrichmentRuleId": "13TNA00000000eH2AQ",
      "DecompRuleId": "13UNA0000004C942AE",
      "FlsrUid": "1FxfIhMq53OIpE9Hzs5w",
      "SourceType": "Attribute",
      "SourceAttributeId": "0tjNA0000004CylYAE",
      "TargetType": "Attribute",
      "TargetAttributeId": "0tjNA0000004CyIYAU",
      "CalculationMethod": "Ad-verbatim",
      "ExecutionStatus": "SUCCESS"
    },
    {
      "EnrichmentRuleId": "13TNA00000000eI2AQ",
      "DecompRuleId": "13UNA0000004C942AE",
      "FlsrUid": "1FxfIhMq53OIpE9Hzs5w",
      "SourceType": "Attribute",
      "SourceAttributeId": "0tjNA0000004CyqYAE",
      "TargetType": "Attribute",
      "TargetAttributeId": "0tjNA0000004CyhYAE",
      "CalculationMethod": "Ad-verbatim",
      "ExecutionStatus": "SUCCESS"
    },
    {
      "EnrichmentRuleId": "13TNA00000000eJ2AQ",
      "DecompRuleId": "13UNA0000004C942AE",
      "FlsrUid": "1FxfIhMq53OIpE9Hzs5w",
      "SourceType": "Attribute",
      "SourceAttributeId": "0tjNA0000004CzPYAU",
      "TargetType": "Attribute",
      "TargetAttributeId": "0tjNA0000004CymYAE",
      "CalculationMethod": "Ad-verbatim",
      "ExecutionStatus": "SUCCESS"
    },
    {
      "EnrichmentRuleId": "13TNA00000000eK2AQ",
      "DecompRuleId": "13UNA0000004C942AE",
      "FlsrUid": "1FxfIhMq53OIpE9Hzs5w",
      "SourceType": "Attribute",
      "SourceAttributeId": "0tjNA0000004CzUYAU",
      "TargetType": "Attribute",
      "TargetAttributeId": "0tjNA0000004CyrYAE",
      "CalculationMethod": "Ad-verbatim",
      "ExecutionStatus": "SUCCESS"
    },
    {
      "EnrichmentRuleId": "13TNA00000000eL2AQ",
      "DecompRuleId": "13UNA0000004C942AE",
      "FlsrUid": "1FxfIhMq53OIpE9Hzs5w",
```

```
        "SourceType": "Attribute",
        "SourceAttributeId": "0tjNA0000004CzZYAU",
        "TargetType": "Attribute",
        "TargetAttributeId": "0tjNA0000004CywYAE",
        "CalculationMethod": "Ad-verbatim",
        "ExecutionStatus": "SUCCESS"
      },
      {
        "EnrichmentRuleId": "13TNA00000000eM2AQ",
        "DecompRuleId": "13UNA0000004C942AE",
        "FlsrUid": "1FxfIhMq53OIpE9Hzs5w",
        "SourceType": "Attribute",
        "SourceAttributeId": "0tjNA0000004D0dYAE",
        "TargetType": "Attribute",
        "TargetAttributeId": "0tjNA0000004Cz1YAE",
        "CalculationMethod": "Ad-verbatim",
        "ExecutionStatus": "ERROR_DATA_TYPE_MISMATCH"
      },
      {
        "EnrichmentRuleId": "13TNA00000000eN2AQ",
        "DecompRuleId": "13UNA0000004C942AE",
        "FlsrUid": "1FxfIhMq53OIpE9Hzs5w",
        "SourceType": "Attribute",
        "SourceAttributeId": "0tjNA0000004D0iYAE",
        "TargetType": "Attribute",
        "TargetAttributeId": "0tjNA0000004CzQYAU",
        "CalculationMethod": "Ad-verbatim",
        "ExecutionStatus": "ERROR_DATA_TYPE_MISMATCH"
      },
      {
        "EnrichmentRuleId": "13TNA00000000eO2AQ",
        "DecompRuleId": "13UNA0000004C942AE",
        "FlsrUid": "1FxfIhMq53OIpE9Hzs5w",
        "SourceType": "Attribute",
        "SourceAttributeId": "0tjNA0000004D18YAE",
        "TargetType": "Attribute",
        "TargetAttributeId": "0tjNA0000004CzVYAU",
        "CalculationMethod": "Ad-verbatim",
        "ExecutionStatus": "ERROR_DATA_TYPE_MISMATCH"
      },
      {
        "EnrichmentRuleId": "13TNA00000000eP2AQ",
        "DecompRuleId": "13UNA0000004C942AE",
        "FlsrUid": "1FxfIhMq53OIpE9Hzs5w",
        "SourceType": "Attribute",
        "SourceAttributeId": "0tjNA0000004D1QYAU",
        "TargetType": "Attribute",
        "TargetAttributeId": "0tjNA0000004CzaYAE",
        "CalculationMethod": "Ad-verbatim",
        "ExecutionStatus": "ERROR_DATA_TYPE_MISMATCH"
      }
    ],
    "OrderId": "801NA000000XKPUYA4",
    "SubmitMode": "Synchronous",
```

```
        "UniqueRequestId": "d00c2aa7-56b0-411a-9b51-83d9fe2e4440",
       "ContexId": "a0ca3c2296c82ad071a5efa83e8974793910f44ebddb21d37f007ce4889b65d7",

        "DesActionSpecDevName": "DroDecompEnrichmentAction",
        "ContextDefinition": "SalesTransactionContext__stdctx"
      },
      "additionalFilter": "undef",
      "applicationLogDate": "Thu Mar 28 12:36:25 GMT 2024",
      "applicationSubtype": "DroDcmp",
      "applicationType": "7",
      "explainabilitySpecName": "DroDecompEnrichmentAction",
      "isChunked": false,
      "name": "DroDecompEnrichmentAction",
      "primaryFilter": "801NA000000XKPUYA4",
      "processType": "DcmpEnrich",
      "secondaryFilter": "1FxfIhMq53OIpE9Hzs5w",
      "uniqueIdentifier": "0bd42942-12b2-4c2e-9f10-b77ee85fd20b"
    }
  ],
  "queryMore": ""
}
```

Get logs for plan composition

This resource example includes sample query parameters to retrieve the logs for plan composition.

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/decision-explainer/action-logs?applicationSubtype=DroDcmp&applicationType=7&processType=DcmpStep&primaryFilter=801xx000003XAG
```

This example shows the sample response. The `actionLog` property contains the action logs.

```
{
  "actionLogs": [
    {
      "actionContextCode": "801NA000000XKPeYAO",
      "actionLog": {
        "PlanCompositionStatus": "",
        "PlanCompositionStatusMessage": "",
        "CandidateProductFulfillmentScenarios": [
          {
            "ProductFulfillmentScenarioId": "1axNA0000004C93YAE",
            "ProductId": "01tNA0000007NP5YAM",
            "FulfillmentStepDefinitionGroupId": "13oNA0000004CARYA2",
            "LineItemIds": [
              "802NA000002lWjOYAU"
            ],
            "ContextRulesetId": "9QwNA0000000CXq0AM"
          }
        ],
        "SelectedProductFulfillmentScenarios": [
          null
        ],
        "SelectedProductFulfillmentScenariosByOli": [
          {
            "OliId": "0a4NA000003HYbFYAW",
            "ProductFulfillmentScenarios": [
```

```
              null
            ]
          },
          {
            "OliId": "0a4NA000003HYbXYAW",
            "ProductFulfillmentScenarios": [
              null
            ]
          },
          {
            "OliId": "0a4NA000003HYbIYAW",
            "ProductFulfillmentScenarios": [
              null
            ]
          },
          {
            "OliId": "802NA000002lWjOYAU",
            "ProductFulfillmentScenarios": [
              null
            ]
          }
        ],
        "PlanId": "13VNA000000OQ9W2AW",
        "FulfillmentStepCreationStatus": "FulfillmentStepsCreated",
        "FulfillmentStepDependencyCreationStatus": "FulfilmentStepDepCreated",
        "OrderId": "801NA000000XKPeYAO",
        "SubmitMode": "Synchronous",
        "UniqueRequestId": "b9ac5971-0d71-45c6-a18e-f472976eaeae",
       "ContextId": "1422049ffa3cc42d8c060b9a7732be360bd62a346652cda9ab6e5fd54cd03eca",

        "DesActionSpecDevName": "DroPlanCompStepsAction",
        "ContextDefinition": "SalesTransactionContext__stdctx"
      },
      "additionalFilter": "undef",
      "applicationLogDate": "Fri Mar 29 10:02:27 GMT 2024",
      "applicationSubtype": "DroPcmp",
      "applicationType": "7",
      "explainabilitySpecName": "DroPlanCompStepsAction",
      "isChunked": false,
      "name": "DroPlanCompStepsAction",
      "primaryFilter": "801NA000000XKPeYAO",
      "processType": "PcmpSteps",
      "secondaryFilter": "undef",
      "uniqueIdentifier": "bb532c9b-a88f-4add-8cd8-bd385359b1ad"
    }
  ],
  "queryMore": ""
}
```


Capture logs for an order

This resource example includes sample query parameters to retrieve the logs for an order.

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/decision-explainer/action-logs?actionContextCode=801DU0000000QBJYA0&applicationType=7
```

This example shows the sample response. The `actionLog` property contains the action logs.


```
{
  "actionLogs": [
    {
      "actionContextCode": "801xx000003GZSxAAO",
      "actionLog":
[redacted]

      "additionalFilter": "undef",
      "applicationLogDate": "Tue May 07 13:05:28 GMT 2024",
      "applicationSubtype": "DroDcmp",
      "applicationType": "7",
      "explainabilitySpecName": "DroDecompEnrichmentAction",
      "isChunked": false,
      "name": "DroDecompEnrichmentAction",
      "primaryFilter": "801xx000003GZSxAAO",
      "processType": "DcmpEnrich",
      "secondaryFilter": "qrh0Z3xv6buew1BW0wdn",
      "uniqueIdentifier": "768f7d03-6770-4939-980f-a6f42ad07cde"
    },
    {
      "actionContextCode": "801xx000003GZSxAAO",
      "actionLog":
[redacted]

      "additionalFilter": "undef",
      "applicationLogDate": "Tue May 07 13:05:28 GMT 2024",
      "applicationSubtype": "DroDcmp",
      "applicationType": "7",
      "explainabilitySpecName": "DroDecompEnrichmentAction",
      "isChunked": false,
      "name": "DroDecompEnrichmentAction",
      "primaryFilter": "801xx000003GZSxAAO",
      "processType": "DcmpEnrich",
      "secondaryFilter": "hjow5L3YcntZsnNuEd5n",
      "uniqueIdentifier": "7ce652e3-81a0-4b2d-a072-88b012c83864"
    },
    {
      "actionContextCode": "801xx000003GZSxAAO",
      "actionLog":
[redacted]

      "additionalFilter": "undef",
      "applicationLogDate": "Tue May 07 13:05:32 GMT 2024",
      "applicationSubtype": "DroDcmp",
      "applicationType": "7",
      "explainabilitySpecName": "DroDecompScopeAction",
      "isChunked": false,
      "name": "DroDecompScopeAction",
      "primaryFilter": "801xx000003GZSxAAO",
      "processType": "DcmpScp",
      "secondaryFilter": "undef",
      "uniqueIdentifier": "9f633a63-67d0-4f90-b4dc-c368defe8b44"
    },
```

```
    {
      "actionContextCode": "801xx000003GZSxAAO",
      "actionLog":
[redacted]

      "additionalFilter": "undef",
      "applicationLogDate": "Tue May 07 13:05:33 GMT 2024",
      "applicationSubtype": "DroPcmp",
      "applicationType": "7",
      "explainabilitySpecName": "DroPlanCompStepsAction",
      "isChunked": false,
      "name": "DroPlanCompStepsAction",
      "primaryFilter": "801xx000003GZSxAAO",
      "processType": "PcmpSteps",
      "secondaryFilter": "undef",
      "uniqueIdentifier": "47a956a6-ae52-4e4a-843b-08b404beb52a"
    },
    {
      "actionContextCode": "801xx000003GZSxAAO",
      "actionLog":
[redacted]

      "additionalFilter": "undef",
      "applicationLogDate": "Tue May 07 13:06:30 GMT 2024",
      "applicationSubtype": "DroSubmit",
      "applicationType": "7",
      "explainabilitySpecName": "DroSubmitAction",
      "isChunked": false,
      "name": "DroSubmitAction",
      "primaryFilter": "801xx000003GZSxAAO",
      "processType": "DroSubmit",
      "secondaryFilter": "undef",
      "uniqueIdentifier": "278caec4-680c-4df8-a1b2-608e05e44baa"
    }
  ],
  "queryMore": ""
}
```


Submit Sales Transaction Action

Initiate the fulfillment process of any sales transaction such as a quote, an order, or an order summary.

Pass the ID of the sales transaction that must be fulfilled to this invocable action. The fulfillment process includes these steps.

- Intake process
- Fulfillment orchestration

The intake process happens synchronously or asynchronously, which is specified by using the `intakeRequestType` input parameter. You can also specify the priority for the execution of the fulfillment process, which is specified by using the `fulfillmentPriority` parameter.

This action is available in API version 63.0 and later.

Special Access Rules

The Submit Sales Transaction action is available in Enterprise, Unlimited, and Developer Editions of Revenue Cloud. See the required permissions to access and call this invocable action.

Supported REST HTTP Methods

URI

`/services/data/v63.0/actions/standard/submitSalesTransaction`

Formats

JSON, XML

HTTP Methods

POST

Authentication

`Authorization: Bearer token`

Input

Input	Details
`fulfillmentAdapter`	**Type** string **Description** Type of fulfillment adapter. Valid value is: • `StandardOrder`
`fulfillmentPriority`	**Type** string **Description** Priority to fulfill the sales transaction. Valid values are: • `High` • `Bulk` • `Default`
`hydratedContextId`	**Type** string **Description** ID of the hydrated context.
`intakeRequestType`	**Type** string

Input	Details
	Description Type of request to process the intake. Valid values are: • `Synchronous` • `Asynchronous`
`priorityLimitAction`	**Type** string **Description** Type of action to perform when the priority limit is reached. Valid values are: • `Reject` • `Downgrade` This parameter is applicable only when the `fulfillmentPriority` parameter is specified.
`salesTransactionId`	**Type** id **Description** Required. ID of the sales transaction to submit.

Output

Output	Details
`errorCode`	**Type** string **Description** Code indicating the type of error.
`fulfillmentPlanId`	**Type** id **Description** ID of the composed fulfillment plan.
`requestedFulfillmentPriority`	**Type** string **Description** Priority to fulfill the sales transaction. Valid values are: • `High` • `Bulk`

Output	Details
	• `Default`
`requestId`	**Type** string **Description** Request ID of the invocation.
`resolvedFulfillmentPriority`	**Type** string **Description** Resolved priority to fulfill the sales transaction.
`submitStatus`	**Type** string **Description** Submit status of the invocation.
`usedContextId`	**Type** string **Description** ID of the used context that updates the decomposition process.

Example

POST

This sample request is for the Submit Sales Transaction action.

```
{
  "inputs": [
    {
      "salesTransactionId": "801DV000000CbIPYA0",
      "intakeRequestType": "Synchronous",
      "fulfillmentAdapter": "StandardOrder",
      "fulfillmentPriority": "Default",
      "priorityLimitAction": "Reject"
    }
  ]
}
```

This sample response is for the Submit Sales Transaction action.

```
[
  {
    "actionName": "submitSalesTransaction",
    "errors": null,
```

```
    "invocationId": null,
    "isSuccess": true,
    "outcome": null,
    "outputValues": {
      "submitStatus": "SUCCESS",
      "resolvedFulfillmentPriority": "Default",
      "requestedFulfillmentPriority": "Default",
     "usedContextId": "0abc8db32b30d09c5051e4561f0b39d938a3bd8db4ccb13e04d41019e427211d",

      "requestId": "927f72b7-85e0-4b5d-b92e-c265f41898f0",
      "fulfillmentPlanId": "13VDV00000008M92AI"
    },
    "sortOrder": -1,
    "version": 1
  }
]
```


Dynamic Revenue Orchestrator Metadata API Types

Metadata API enables you to access some types and feature settings that you can customize in the user interface.

Flow for Dynamic Revenue Orchestrator
Represents the metadata associated with a flow. With Flow, you can create an application that navigates users through a series of screens to query and update records in the database. You can also execute logic and provide branching capability based on user input to build dynamic applications.

DynamicFulfillmentOrchestratorSettings
Represents the settings for Dynamic Revenue Orchestrator.

SEE ALSO:

Metadata API Developer Guide: Understanding Metadata API

Flow for Dynamic Revenue Orchestrator

Represents the metadata associated with a flow. With Flow, you can create an application that navigates users through a series of screens to query and update records in the database. You can also execute logic and provide branching capability based on user input to build dynamic applications.

FlowActionCall

Dynamic Revenue Orchestrator exposes additional actionType values for the FlowActionCall Metadata type.

Field Name	Field Type	Description
`actionType`	InvocableActionType (enumeration of type string)	Required. The action type. Additional valid values only for Dynamic Revenue Orchestrator include:

Field Name	Field Type	Description
		• `submitOrder`—Submit an order to Dynamic Revenue Orchestrator (DRO) for fulfillment.

DynamicFulfillmentOrchestratorSettings

Represents the settings for Dynamic Revenue Orchestrator.

Parent Type and Manifest Access

This type extends the Metadata metadata type and inherits its `fullName` field.

In the package manifest, all the settings metadata types for the org are accessed using the "Settings" name. See Settings for more details.

File Suffix and Directory Location

`DynamicFulfillmentOrchestratorSettings` values are stored in the `DynamicFulfillmentOrchestratorSettings.settings` file in the `settings` folder. The `.settings` files are different from other named components, because there is only one settings file for each settings component.

Version

DynamicFulfillmentOrchestratorSettings components are available in API version 61.0 and later.

Fields

Field Name	Description
`enableDFOFallout`	**Field Type** boolean **Description** Indicates whether to enable fallout management to handle fallouts and retry policies (`true`) or not (`false`). The default value is `false`. See Turn On Feature to Manage Fallout.
`enableDFOJeopardy`	**Field Type** boolean **Description** Indicates whether to enable management of Service Level Agreements (`true`) or not (`false`). The default value is `false`. See Turn On Feature to Manage Service Level Agreements.
`enableDFOPref`	**Field Type** boolean

Field Name	Description
	Description Indicates whether to enable features of Dynamic Revenue Orchestrator (`true`) or not (`false`). The default value is `false`. See Turn On Dynamic Revenue Orchestrator.
`enableDROFutureDatedTasks`	**Field Type** boolean **Description** Indicates whether to enable the Future Dated Steps feature (`true`) or not (`false`). The default value is `false`. See Enable Future Dated Steps. Available in API version 63.0 and later.
`enableDROTaskSource`	**Field Type** boolean **Description** Indicates whether to link a Salesforce manual task to a fulfillment step source (`true`) or not (`false`). The default value is `false`. See Enable the Linking of Task to Step Source. Available in API version 63.0 and later.

Declarative Metadata Sample Definition

The following is an example of a DynamicFulfillmentOrchestratorSettings component.

```
<DynamicFulfillmentOrchestratorSettings xmlns="http://soap.sforce.com/2006/04/metadata">
     <enableDFOPref>true</enableDFOPref>
     <enableDFOFallout>true</enableDFOFallout>
     <enableDFOJeopardy>true</enableDFOJeopardy>
     <enableDROFutureDatedTasks>true</enableDROFutureDatedTasks>
     <enableDROTaskSource>true</enableDROTaskSource>
</DynamicFulfillmentOrchestratorSettings>
```

The following is an example `package.xml` that references the previous definition.

```
<?xml version="1.0" encoding="UTF-8"?>
<Package xmlns="http://soap.sforce.com/2006/04/metadata">
    <types>
        <members>DynamicFulfillmentOrchestrator</members>
        <name>Settings</name>
    </types>
    <version>63.0</version>
</Package>
```

Wildcard Support in the Manifest File

The wildcard character `*` (asterisk) in the `package.xml` manifest file doesn't apply to metadata types for feature settings. The wildcard applies only when retrieving all settings, not for an individual setting. For details, see Settings. For information about using the manifest file, see Deploying and Retrieving Metadata with the Zip File.

Callouts in Dynamic Revenue Orchestrator

Use callout step types to make HTTP calls to an external system. Callout step types use the Data Consumption Framework and Process Type Integration for interface management and communication with external systems.

Use callout step types to implement these features.

- Endpoint configuration
- Authentication parameters
- Default sync and async implementation
- Extensions or Apex Interfaces
- Error handling and fallout management
- Request and response logging
- Timeouts

Dynamic Revenue Orchestrator supports these types of callouts.

Type	Description
Default Order Fulfillment Provider	Use this callout provider to use a fixed payload structure to request an order. You can also define Omnistudio Integration Procedures for request transformation and response handling.
Apex Type Provider	Use this callout provider to implement custom integration scenarios via the `industriesintegrationfwk.ProcessIntegrationProvider` Apex interface.
External Services Defined Provider	Use this callout provider to implement custom integration scenarios via external services through the OpenAPI specification. You can also define Omnistudio Integration Procedures for request transformation and response handling. To use this callout, the Omnistudio Admin and Omnistudio Runtime permissions are required.

Callouts support these integration patterns.

Type	Description
Sync	The callout step is completed when the external system returns a response.
Async	The callout step is set to `In Progress` when the external system returns the acknowledgment response with 202 HTTP code. The system waits until the external system sends a callback response to complete the callout step.

Configure Callout Settings

Before you set up a callout provider, configure the callout settings. The settings include the creation of a named credential and an external credential, the creation of an integration definition, and the configuration of a fulfillment step definition.

Default Order Fulfillment Provider

The Default Order Fulfillment Provider or `CalloutIntegrationProvider` is a provider for the order fulfillment usage type. A Fulfillment Designer can configure this provider.

Apex Type Provider

Implement custom integration logic via Apex by using the Apex Type Provider. This provider requires an Apex Integration Developer to implement a custom Apex adapter interface.

External Services Defined Provider

Generate interface contract and Apex types by using external services and Open API compatible schema.

SEE ALSO:

Salesforce Help: Data Consumption Framework

Configure Callout Settings

Before you set up a callout provider, configure the callout settings. The settings include the creation of a named credential and an external credential, the creation of an integration definition, and the configuration of a fulfillment step definition.

Meet these prerequisites before you configure the callout settings.

- Configure named and external credentials to define access to an external system. Specify a named credential as the callout endpoint and an external credential to configure the authentication protocol.

  See Create Named Credentials and External Credentials.

- Create integration definitions to connect Salesforce with an external system. Integration definitions use APIs to perform operations in both Salesforce and the external system. You can create Apex Defined, External Services Defined, or Standard integration definitions.

  See Create an Integration Definition.

- Define a fulfillment step with `Callout` as the step type. Additionally, set the integration definition name and integration user.

  See Callout Fulfillment Step.

EDITIONS

Available in: **Developer**, **Enterprise**, and **Unlimited** Editions

USER PERMISSIONS

To create authenticated callouts:

- External Credentials Principal Access Permission

Default Order Fulfillment Provider

The Default Order Fulfillment Provider or `CalloutIntegrationProvider` is a provider for the order fulfillment usage type. A Fulfillment Designer can configure this provider.

The Default Order Fulfillment Provider includes these features.

- Predefined payload with order and fulfillment order data, including attribute values
- Fixed integration parameters such as timeouts, credentials, and path
- Modified requests and responses via Omnistudio Integration Procedures
- Asynchronous interaction pattern

- Request and response logging
- Errors via fallout rules

To configure the callout settings for Default Order Fulfillment Provider, see Configuration Steps.

Error Handling

To verify if the callout request was successful, check the `status` value in the payload. If the status is undefined, then these HTTP codes indicate a successful response.

- `200`
- `201`
- `202`
- `203`
- `204`
- `205`
- `206`
- `302`
- `304`

If the request isn't successful, then the fulfillment step state is marked as `Fatally Failed`.

Integration Definition Configurations

You can configure these additional features for the integration definition.

- Select the **Save the request and response as attachments to the record** checkbox for the integration definition to save request and response payloads as attachments to the Integration Provider Execution record. Content publish limits apply when saving request and response payloads as attachments. Use Shield Platform Encryption for secure storage of sensitive information.
- Define Input and Output Processors for the Integration Definition for the pre-processing of the default order fulfillment request before you send the request to an external system. See Omnistudio Integration Procedures.

See Create an Integration Definition.

Apex Type Provider

Implement custom integration logic via Apex by using the Apex Type Provider. This provider requires an Apex Integration Developer to implement a custom Apex adapter interface.

Use a custom Apex adapter to:

- Include and use integration parameters in adapter implementations such as timeouts, credentials, and path.
- Generate and transform requests and responses via Omnistudio Integration Procedures or Extract, Transform, Load (ETL) libraries.
- Enable asynchronous interaction pattern.
- Enable request and response logging by implementing a specific Apex interface.
- Handle errors, and integrate with fallout rules.

To configure the callout settings for Apex Type Provider, see Configuration Steps.

Integration Adapter Implementation

Implement the `industriesintegrationfwk.ProcessIntegrationProvider` Apex interface. Use the `industriesintegrationfwk.ApexProviderAttr` class to define and access the attribute values in the integration provider definition.

```
global with sharing class DROSampleOrderAdapter implements
industriesintegrationfwk.ProcessIntegrationProvider {

    // Named credential attribute
    private static final String MOCK_CALLOUT_NAMED_CREDENTIAL =
'callout:DROOrderInterfaceNamedCred';
    private final static Integer TIMEOUT = 10000; // Request time out in milliseconds, or
 can be defined on Integration Definition as an Attribute


    private static final industriesintegrationfwk.ApexProviderAttr NAMED_CRED_ATTR = new
industriesintegrationfwk.ApexProviderAttr('Named Credential',
    'Named_Credential', 'DROOrderInterfaceNamedCred', true, 'String');
    private static final industriesintegrationfwk.ApexProviderAttr ENDPOINT_ATTR = new
industriesintegrationfwk.ApexProviderAttr('Endpoint URI',
    'Endpoint_URI', '/v1/orderitems/', true, 'String');


    /**
     * @param requestGuid                 Request GUID provided by the client
     * @param inputRecordId              Input Record ID - value is taken from Fulfillment
 Step > Fulfillment Step Source > Source Line Item
     * @param payload                    Payload to be passed to the Provider Class (empty
 in DRO)
     * @param attributes                  Map of config attributes
     * @return IntegrationCalloutResponse Response sent to the client
    */
   global static industriesintegrationfwk.IntegrationCalloutResponse executeCallout(String
 requestGuid, String inputRecordId, String payload, Map<String, Object> attributes) {
        // create request
        String msgBody = '{\"message\":\"Hello\",'
            + '\n\"requestGuid\":\"' + requestGuid + '\",\n'
            + '\n\"inputRecordId\":\"' + inputRecordId + '\",\n'
            + '\n\"payload\":\"' + payload + '\"}'
            ;

        String endpointUri = (String) attributes.get('Endpoint_URI');

        // make a call
        HttpResponse response = makeCallout(endpointUri, msgBody);
        // process response and pass details to the DRO Callout Step processor by using
IntegrationCalloutResponse(isSuccess, ResponseCode, ErrorMessage)

         industriesintegrationfwk.IntegrationCalloutResponse integrationCalloutResponse =
 handleResponse(response);

        return integrationCalloutResponse;
    }
```

```
    // define configurable attriobutes
    global static List<industriesintegrationfwk.ApexProviderAttr> getProviderAttributes()
 {
        List<industriesintegrationfwk.ApexProviderAttr> defaults = new
List<industriesintegrationfwk.ApexProviderAttr>();
        defaults.add(NAMED_CRED_ATTR);
        defaults.add(ENDPOINT_ATTR);
        // add any attributes such as endpoint, timeout, interface, and params
        // ...
        return defaults;
    }

    // Call Mock Service API.
    private static HttpResponse makeCallout(String endpointUri, String msgBody) {

        // Construct the request object
        String endPoint = MOCK_CALLOUT_NAMED_CREDENTIAL + endpointUri;
        HttpRequest request = new HttpRequest();
        request.setMethod('POST');
        request.setHeader('Content-Type', 'application/json');
        request.setHeader('Accept', 'application/json');
        request.setEndpoint(endPoint);
        request.setTimeout(TIMEOUT);
        request.setBody(msgBody);

        // Send request
        HttpResponse response = new Http().send(request);
        return response;
    }
}
```

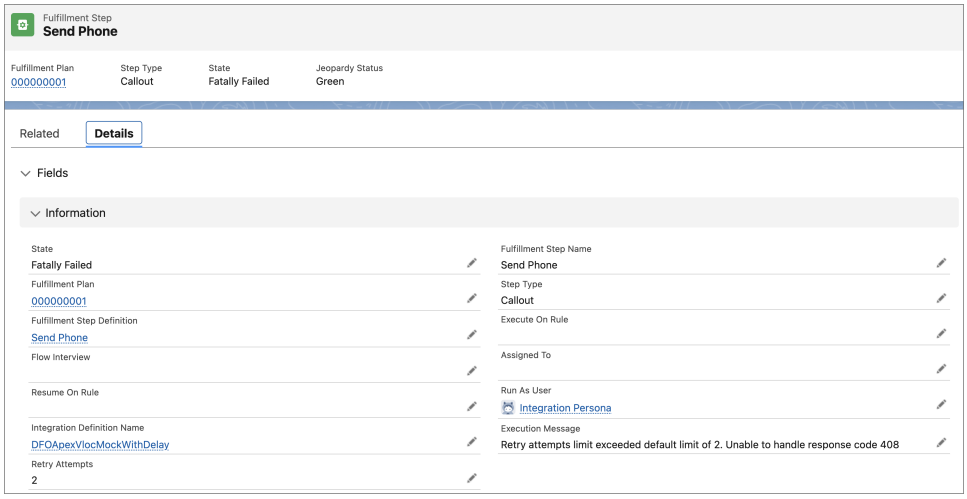
Error Handling

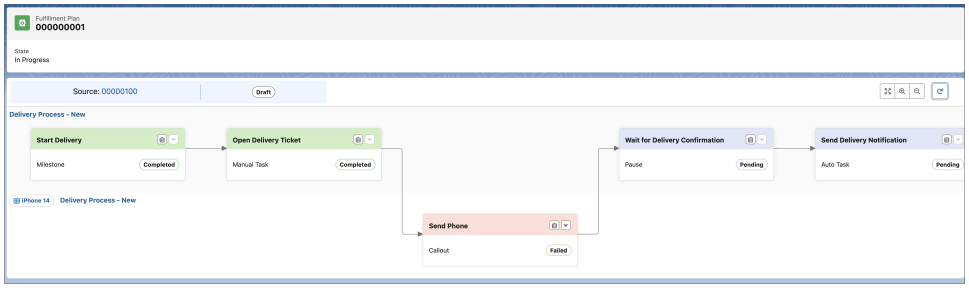
This sample shows how to pass errors from response to orchestration via the `IntegrationResponse` interface in the Apex provider implementation. See ProcessIntegrationProvider Interface.

```
public static industriesintegrationfwk.IntegrationCalloutResponse handleResponse(HttpResponse
 response) {

        industriesintegrationfwk.IntegrationCalloutResponse integrationCalloutResponse;

        Map<String, Object> responseGroup = getResponseGroupAfterCallout(response);

        if(response.getStatusCode() == 200)
        {
            // SUCCESS
            integrationCalloutResponse = new
industriesintegrationfwk.IntegrationCalloutResponse(true);
                    integrationCalloutResponse.setResponseCode(response.getStatusCode());

               integrationCalloutResponse.setReturnValue(responseGroup);

        } else {
            //FAILURE - pass error to DRO Fallout Handling to apply Retry Policies
```

```
            integrationCalloutResponse = new
industriesintegrationfwk.IntegrationCalloutResponse(false);
            integrationCalloutResponse.setResponseCode(response.getStatusCode());
            integrationCalloutResponse.setReturnValue(responseGroup);
            integrationCalloutResponse.setErrorMessage('Unable to process request.');
        }
        return integrationCalloutResponse;
        }

    // Process Response payLoad
   private static Map<String, Object> getResponseGroupAfterCallout(HttpResponse response)
 {
        Map<String, Object> responseGroup = new Map<String, Object>();
        if (response.getStatusCode() == 200) {
            responseGroup.put('isSuccess', true);
        } else {
            responseGroup.put('isSuccess', false);
        }
        responseGroup.put('response', getResponseMap(response.getBody()));
        return responseGroup;
    }

    // Convert response string into Map
    private static Map<String,Object> getResponseMap(String responseBody) {
        try {
            Map<String,Object> responseBodyMap = (Map<String,Object>)
JSON.deserializeUntyped(responseBody);
            return responseBodyMap;
        } catch (Exception e) {
            Map<String, Object> responseMap = new Map<String,Object>();
            responseMap.put('response', responseBody);
            return responseMap;
        }
    }
```

The fulfillment step state changes to `Fatally Failed`, and the error message is saved in the `Execution Message` field.





Apex Advanced Interface Implementation

To use advanced features such as logging, the Apex provider must implement the Apex Provider Advanced interface. You must use the HttpBaseProvider client for HTTP requests.

This example shows a sample of the `ProcessIntegrationProviderAdvanced` Apex interface implementation.

```
global with sharing class DFOApexVlocMockWithDelay
implements industriesintegrationfwk.ProcessIntegrationProviderAdvanced {
...
global static industriesintegrationfwk.IntegrationCalloutResponse executeCallout(String
requestGuid, String inputRecordId, String payload, Map<String, Object> attributes,
industriesintegrationfwk.HttpBaseProvider httpProvider) {
...
String endPoint = MOCK_CALLOUT_NAMED_CREDENTIAL + '/delay/5';
HttpRequest request = new HttpRequest();
request.setMethod('POST');
request.setHeader('Content-Type', 'application/json');
request.setHeader('Accept', 'application/json');
request.setEndpoint(endPoint);
request.setTimeout(TIMEOUT);
request.setBody(msgBody);

// Send request
//HttpResponse response = httpProvider.httpCallout(request);
HttpResponse response = new Http().send(request);
```

Integration Definition Configuration

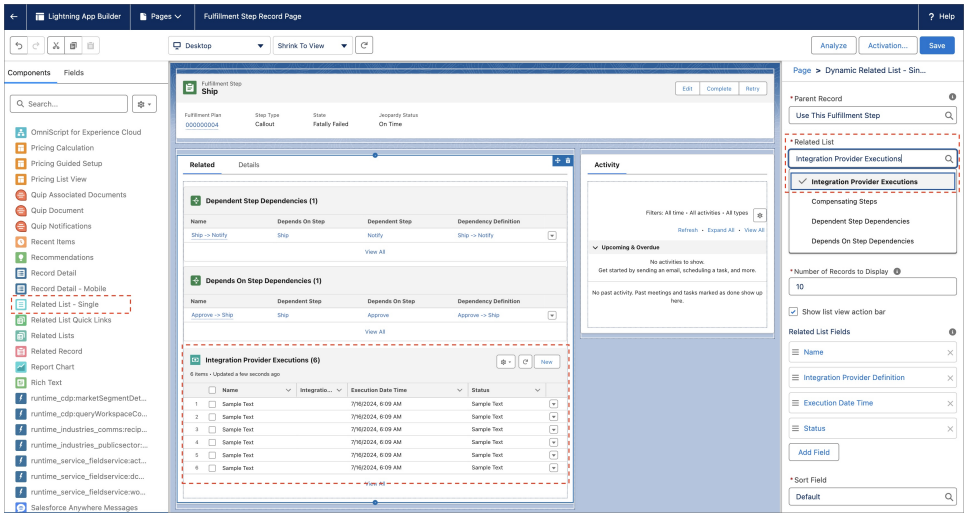
Select the **Save the request and response as attachments to the record** checkbox for the integration definition to save request and response payloads as attachments to the Integration Provider Execution record. Content publish limits apply when saving request and response payloads as attachments. Use Shield Platform Encryption for secure storage of sensitive information.

See Create an Integration Definition.

Log Records

Integration Provider execution records are created on every request and associated with the fulfillment step. The referenced object identifier is set to order item or fulfillment order line item ID.

To view the Integration Provider execution records, configure the Fulfillment Step record page to include the Integration Provider Executions related list by using Lightning App Builder. See Lightning App Builder.



External Services Defined Provider

Generate interface contract and Apex types by using external services and Open API compatible schema.

To configure the callout settings for External Services Defined Provider, see Configuration Steps.

As an integration specialist user or admin user, perform these steps.

- Set up an external service and actions.
- In the external service definition, include integration parameters such as error codes, credentials, and path.

External Service

Use external services for outbound integrations from Salesforce by using low-code, process-based integrations to enhance your Apex integrations. See External Services.

Integration Definition Configurations

You can configure these additional features for the integration definition.

- Select the **Save the request and response as attachments to the record** checkbox for the integration definition to save request and response payloads as attachments to the Integration Provider Execution record. Content publish limits apply when saving request and response payloads as attachments. Use Shield Platform Encryption for secure storage of sensitive information.
- Define Input and Output Processors for the Integration Definition for the pre-processing of the default order fulfillment request before you send the request to an external system. See Omnistudio Integration Procedures.

See Create an Integration Definition.

Step Definition

Set the created integration definition on the Step Definition record with `Callout` as the step type.

CHAPTER 9 Billing

In this chapter ...

- Billing Standard Objects
- Billing Platform Events
- Billing Standard Invocable Actions
- Billing Business APIs
- Billing Apex Reference
- Billing Metadata API Types

Automate processes related to billing, credit application, and invoice generation. Generate billing schedules by using context service, and create invoices from a billing schedule.

SEE ALSO:

Salesforce Help: Assign Permissions to Access Billing Features

EDITIONS

Available in: Lightning Experience

Available in: **Enterprise**, **Unlimited**, and **Developer** Editions

Billing Standard Objects

The Billing data model provides objects and fields to manage billing and tax configurations, credit memos, and invoices, and accounting periods for legal entities.

AccountBillingAccount

Represents a junction between Account objects and Billing Account objects. This object is available in API version 63.0 and later.

AccountingPeriod

Represents information about a time period for which businesses prepare reports and analyze performance. Each billing transaction is associated with an accounting period. This object is available in API version 62.0 and later.

BillingAccount

Represents the customer's billing controls in the backend system. This object is used to set up and maintain details such as bill frequency, bill format, and autopay details. This object is available in API version 63.0 and later.

BillingBatchScheduler

Represents a scheduled processing job that triggers recurring invoice batch runs. This object is available in API version 62.0 and later.

BillingBatchFilterCriteria

Represents the filter that all eligible billing schedules must satisfy in order to be picked up by an invoice run. This object is available in API version 62.0 and later.

BillingMilestonePlan

Represents a structured approach to invoicing where invoices are scheduled based on predefined milestones. This object is available in API version 63.0 and later.

BillingMilestonePlanItem

Represents a specific billing milestone within the billing milestone plan that’s used to manage and track billing based on the completion of certain deliverables or stages. This object is available in API version 63.0 and later.

BillingPeriodItem

Represents a payment period for an invoice. The billing period item is used to pass billing information to an invoice line. This object is available in API version 62.0 and later.

BillingPolicy

Represents information about a set of billing treatments that define the rules to invoice a customer for an order item. This object is available in API version 62.0 and later.

BillingSchedule

Represents information about the order item that's used in the invoicing process. This object is available in API version 62.0 and later.

BillingScheduleGroup

Represents a consolidated view of all the billing schedules related to the order items generated from one asset, including new orders and amendment orders. This object is available in API version 62.0 and later.

BillingTreatment

Represents information about the billing of an order item. This object is available in API version 62.0 and later.

BillingTreatmentItem

Represents information about allocation of the total amount of an order item to billing schedules throughout the order item's lifecycle. This object is available in API version 62.0 and later.

BsgRelationship

Represents a relationship between billing schedule groups to support bundles where one parent billing schedule group has multiple child billing schedule groups. This object is available in API version 62.0 and later.

CreditMemo

Represents a document that's used to reduce the amount that a buyer owes a seller under the terms of an earlier invoice. This object is available in API version 62.0 and later.

CreditMemoAddressGroup

Represents the storage of the buyer's address information, which is used to determine the tax credit amount for a buyer when a credit memo is issued. This object is available in API version 62.0 and later.

CreditMemoInvApplication

Represents information about the application of a credit memo to an invoice. This object is available in API version 62.0 and later.

CreditMemoLine

Represents the product, service, adjustment, or tax line items included in a credit memo. This object is available in API version 62.0 and later.

CreditMemoLineInvoiceLine

Represents a junction between a credit memo line and an invoice line. This object is available in API version 62.0 and later.

CreditMemoLineTax

Represents tax information of a credit memo line of type `Tax`. This object is available in API version 62.0 and later.

GeneralLedgerAccount

Represents information about the accounting codes, types, and names that are used to store and organize financial transactions. This object is available in API version 63.0 and later.

GeneralLedgerAcctAsgntRule

Represents information about the rule based on which general ledger accounts are assigned to transaction journals that are created for billing transactions. This object is available in API version 63.0 and later.

InvBatchDraftToPostedRun

Represents information about the batch job that posts all invoices with the status as `Draft` that are generated by the invoice batch run associated with the billing schedule. This object is available in API version 62.0 and later.

Invoice

Represents information about a financial document describing the total amount a buyer must pay for provided products or services. This object is available in API version 62.0 and later.

InvoiceAddressGroup

Represents the storage of the buyer's address information. This object is available in API version 62.0 and later.

InvoiceBatchRun

Represents a batch processing job in Billing. During an invoice batch run, all billing schedules that meet the specified criteria are processed, resulting in the generation of invoices. This object is available in API version 62.0 and later.

InvoiceBatchRunCriteria

Represents a batch processing job and its required criteria in Billing. During an invoice batch run, all billing schedules that meet the specified criteria are processed, resulting in the generation of invoices. This object is available in API version 62.0 and later.

InvoiceBatchRunRecovery

Represents information about the recovery procedure of an invoice batch run. This object is available in API version 62.0 and later.

InvoiceDocument

Represents the PDF document generated for an invoice. This object is available in API version 63.0 and later.

InvoiceLine

Represents the amount that a buyer must pay for a product, service, or fee. Invoice lines are created based on the amount of an order line. This object is available in API version 62.0 and later.

InvoiceLineRelationship

Represents a relationship between invoice line items to support bundles where one parent invoice line has multiple child invoice lines. This object is available in API version 62.0 and later.

InvoiceLineTax

Represents tax information of an invoice line of type `Tax`. This object is available in API version 62.0 and later.

LegalEntity

Represents the way an organization is structured. An organization can be a single legal entity or it can comprise more than one legal entity. This object is available in API version 62.0 and later.

LegalEntyAccountingPeriod

Represents a junction between a legal entity and an accounting period. This object is available in API version 62.0 and later.

PaymentTerm

Represents an agreement between a buyer and a seller about when payment is due for an invoice. This object is available in API version 62.0 and later.

PaymentTermItem

Represents configuration of a payment term. This object is available in API version 62.0 and later.

RevenueTransactionErrorLog

Represents the details of errors that occurred during the processing of a request. The error record persists until a new error with the same category, primary record, and, if necessary, related record occurs. This object is available in API version 62.0 and later.

TaxEngine

Represents information about an instance of a tax engine provider as well as the merchant credentials for that specific instance. This object is available in API version 62.0 and later.

TaxEngineProvider

Represents general information about a service that manages a tax engine. Tax engine providers have a one-to-many relationship with tax engines, where the tax engine record represents a specific configuration of a tax engine that can be assigned to multiple order items. This object is available in API version 62.0 and later.

TaxPolicy

Represents information about a group of tax treatments, where each treatment represents parameters to determine how a particular product is taxed for a transaction line item. Tax policies are related to products, which pass the policy on to the resulting order items and in turn the billing schedules. This object is available in API version 62.0 and later.

TaxTreatment

Represents information about tax calculation by external engines. Each product requires a tax policy to determine whether to apply tax. Each tax policy requires at least one tax treatment. The tax treatments determine how taxable products are taxed. This object is available in API version 62.0 and later.

TransactionJournal

Represents details about billing transactions that are recorded for general ledger accounting purposes. This object is available in API version 63.0 and later.

SEE ALSO:

Object Reference for the Salesforce Platform: Overview of Salesforce Objects and Fields

SOAP API Developer Guide: Introduction to SOAP API

AccountBillingAccount

Represents a junction between Account objects and Billing Account objects. This object is available in API version 63.0 and later.

Supported Calls

`describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search()`

Special Access Rules

You need the Billing Admin permission set, Billing Operations User permission set, or Billing Customer Service User permission set access to this object.

Fields

Field	Details
`IsDefaultBillingAccount`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Indicates whether the account is the default billing account (`true`) or not (`false`). The default value is `false`.

AccountingPeriod

Represents information about a time period for which businesses prepare reports and analyze performance. Each billing transaction is associated with an accounting period. This object is available in API version 62.0 and later.

Supported Calls

`create(), delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), undelete(), update(), upsert()`

Special Access Rules

You need the Accounts Receivables Admin permission set to access this object.

Fields

Field	Details
`EndDate`	**Type** date

Field	Details
	Properties Create, Filter, Group, Sort, Update **Description** Required. The end date of an accounting period.
`FinancialYear`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** Required. The financial year in which an accounting period falls.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed an accounting period indirectly, for example, through a list view or related record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed an accounting period. If this value is null, it's possible that the user only accessed the accounting period or a related list view (`LastReferencedDate`), but not viewed the accounting period itself.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** Required. The auto-generated name of an accounting period. The name is a combination of the accounting period's start date, start month, end date, and end month.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update

Field	Details
	Description Required. The user who owns an Accounting Period record. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
`StartDate`	**Type** date **Properties** Create, Filter, Group, Sort, Update **Description** Required. The start date of an accounting period.
`Status`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. The status of an accounting period. Valid values are: • `Closed` • `Open`

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

AccountingPeriodShare **on page 1684**
Sharing is available for the object.

BillingAccount

Represents the customer's billing controls in the backend system. This object is used to set up and maintain details such as bill frequency, bill format, and autopay details. This object is available in API version 63.0 and later.

Supported Calls

`describeLayout(),describeSObjects(),getDeleted(),getUpdated(),query(),retrieve(),search()`

Special Access Rules

You need the Billing Admin permission set, Billing Operations User permission set, or Billing Customer Service User permission set access to this object.

Fields

Field	Details
`BillingResumptionDate`	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The date when billing for the account is resumed.
`BillingSuspensionDate`	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The date when billing for the account is suspended.

BillingBatchScheduler

Represents a scheduled processing job that triggers recurring invoice batch runs. This object is available in API version 62.0 and later.

Supported Calls

`describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`

Special Access Rules

You need the Billing Admin permission set to access this object.

Fields

Field	Details
`BillingSchedulerName`	**Type** string **Properties** Filter, Group, idLookup, Sort

Field	Details
	Description Required. The name of the scheduler.
`Comments`	**Type** textarea **Properties** Filter, Nillable, Sort **Description** Additional details about the billing batch scheduler.
`CronExpression`	**Type** string **Properties** Filter, Group, Sort **Description** Required. This field determines how often the scheduler recurs.
`EndDate`	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The date when the scheduler stops triggering batch processing jobs.
`FrequencyCadence`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Required. The frequency of the scheduler triggering the invoice batch run. Valid values are: • `Daily`— The scheduled job recurs every day. • `Monthly`— The scheduled job recurs every month. • `Once`— The scheduled job occurs one time and doesn't recur. • `Weekly`— The scheduled job recurs every week.
`FrequencyOptions`	**Type** textarea **Properties** Nillable

Field	Details
	Description This field is a derived field that stores the scheduler configuration.
`JobType`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Required. The type of batch processing job that the scheduler triggers. Valid value is `Invoice` for which the scheduler starts a batch invoice run.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a billing batch scheduler record indirectly, for example, through a list view or related record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a billing batch scheduler record. If this value is null, it's possible that the user only accessed the billing batch scheduler record or a related list view (`LastReferencedDate`), but not viewed the billing batch scheduler record itself.
`NextRunTime`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date and timestamp of the next scheduled batch invoice run in the user's time zone.
`OwnerId`	**Type** reference **Properties** Filter, Group, Sort **Description** Required. The ID of the user who created the scheduler.

Field	Details
	This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
`RecurringSubType`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** The frequency at which the batch processing job recurs when the `FrequencyCadence` field value is set to `Monthly`. Valid values are: • `Every`— The processing job recurs at every instance of the frequency of the value. For example, if the `RecurringSubType` field value is `Every` and the `FrequencyCadence` field value is `Weekly`, then the batch processing job recurs every week. • `SpecificDate`— The scheduler triggers the batch processing job on the selected date. For example, if the selected date is 5, and the `FrequencyCadence` field value is `Monthly`, then the job recurs on the fifth day of each month.
`RecurringType`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** The frequency at which the batch processing job is repeated when the `FrequencyCadence` field value is set to `Weekly`. Valid value is `Every`.
`RecursOn`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** The interval at which the scheduler triggers a batch processing job. If the `FrequencyCadence` field value is `Monthly`, you must select either the specific date or the interval when the schedule triggers the job. Valid values are: • `First`

Field	Details
	• `Fourth` • `Last` • `Second` • `Third` **Example:** To configure the scheduler to trigger the job on the first Monday of the month, set the following fields: • `FrequencyCadence=Monthly` • `RecursOn=First` • `RecursOnDay= Monday`
`RecursOnDate`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The date on which the scheduler triggers a batch processing job. **Example:** To configure the scheduler to trigger the job on the fifth day of the month, set the following fields: • `FrequencyCadence=Monthly` • `RecursOnDate=` 5 **Example:** To configure the scheduler to trigger the job on the second to last day of the month, set the following fields: • `FrequencyCadence=Monthly` • `RecursOnDate=SecondToLast` If you select `Last`, `SecondToLast`, or `ThirdToLast`, the date of the batch processing job varies depending on the number of days in the month. For example, consider `SecondToLast` is selected. If the month has 30 days, such as June, then the batch processing job occurs on the 28th day. If the month has 31 days, such as July, then the batch processing job occurs on the 29th day.
`RecursOnDay`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** The day on which the scheduler triggers a batch processing job. If the `FrequencyCadence` field value is set to `Weekly`, then you must select the day when the scheduler runs. The scheduler recurs every week on the selected day; for example, weekly on Monday. Valid values are:

Field	Details
	• `Sunday` • `Monday` • `Tuesday` • `Wednesday` • `Thursday` • `Friday` • `Saturday`
`RunCriteriaId`	**Type** reference **Properties** Filter, Group, Sort **Description** Required. The ID of the filter criteria that's defined for the invoice batch run or the payment batch run. This field is a polymorphic relationship field. **Relationship Name** RunCriteria **Refers To** InvoiceBatchRunCriteria
`ShouldExcludeWkendAndHldy`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Required. Indicates whether weekends and holidays are excluded from the billing schedule (`true`) or not (`false`). The default value is `false`.
`ShouldStartRunImmediately`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Required. Indicates whether the invoice scheduler must start the run immediately (`true`) or not (`false`). The default value is `false`. Available in API version 63.0 and later.
`StartDate`	**Type** date

Field	Details
	Properties Filter, Group, Sort **Description** Required. The date when the scheduler triggers its first batch processing job.
`StartTime`	**Type** time **Properties** Filter, Sort **Description** Required. The time when the scheduler triggers the batch processing job.
`Status`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Required. The status of the scheduler. Only active schedulers can trigger batch processing jobs. Valid values are: • `Active` • `Canceled` • `Draft` • `Inactive`
`TimeZone`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** The time zone is either the value selected when the run was configured, or it's the user's time zone. The time zone is shown in Greenwich Mean Time (GMT).

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

BillingBatchSchedulerShare **on page 1684**
Sharing is available for the object.

BillingBatchFilterCriteria

Represents the filter that all eligible billing schedules must satisfy in order to be picked up by an invoice run. This object is available in API version 62.0 and later.

Supported Calls

`describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

You need the Billing Admin permission set to access this object.

Fields

Field	Details
`BatchCriteriaId`	**Type** reference **Properties** Filter, Group, Sort **Description** Required. The ID of the batch criteria record. This field is a polymorphic relationship field. **Relationship Name** BatchCriteria **Refers To** InvoiceBatchRunCriteria
`BillingBatchFilterCriteriaNumber`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. The auto-generated reference number for the billing batch filter criteria.
`ColumnEnum`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Required. The column or field to which the filter criteria are applied.

Field	Details
`CriteriaFieldType`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** The data type of the custom or standard criteria field. Valid values are: • `CustomBoolean` • `CustomCurrency` • `CustomDate` • `CustomLookup` • `CustomNumber` • `CustomPercent` • `CustomText` Available in API version 63.0 and later.
`CriteriaSequence`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The order in which the filter criteria are applied on the billing batch.
`InvoiceRunMatchingValue`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** A value to match during an invoice run. This field is useful for filtering the invoices based on specific criteria during the billing process.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed this record indirectly, for example, through a list view or related record.
`LastViewedDate`	**Type** dateTime

Field	Details
	Properties Filter, Nillable, Sort **Description** The timestamp when the current user last viewed this record or list view. If this value is null, and `LastReferenceDate` is not null, the user accessed this record or list view indirectly.
`ObjectName`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Required. The object on which the filter criteria are applied. Valid values are: • `AccountBillingAccount` — Available in API version 63.0 and later. • `BillingAccount` — Available in API version 63.0 and later. • `BillingSchedule` • `BillingScheduleGroup` • `CreditMemo` — Available in API version 63.0 and later. • `CreditMemoInvApplication` — Available in API version 63.0 and later. • `CreditMemoLine` — Available in API version 63.0 and later. • `CreditMemoLineInvoiceLine` — Available in API version 63.0 and later. • `CreditMemoLineTax` — Available in API version 63.0 and later. • `Invoice` — Available in API version 63.0 and later. • `InvoiceLine` — Available in API version 63.0 and later. • `InvoiceLineTax` — Available in API version 63.0 and later.
`Operation`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Required. The type of comparison or logical operation to be performed on the specified column or field. Valid values are: • `Contains` — Available in API version 63.0 and later. • `Equals` • `GreaterThan` — Available in API version 63.0 and later. • `GreaterThanOrEqualTo` — Available in API version 63.0 and later. • `InList`

Field	Details
	• `LessThan` — Available in API version 63.0 and later. • `LessThanOrEqualTo` • `NotEquals` — Available in API version 63.0 and later. • `OfType` • `StartsWith` — Available in API version 63.0 and later.
`OwnerId`	**Type** reference **Properties** Filter, Group, Sort **Description** Required. The ID of the user who created the billing batch filter criteria. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
`PaymentRunMatchingValue`	**Type** picklist **Properties** Filter, Group, Nillable, Sort **Description** A value to match during a payment run. This field is useful for filtering the payments based on specific criteria during the billing process. This field is visible but isn't used in API version 62.0.
`StandardCriteriaField`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The name of the standard field for the object specified in the `ObjectName` field. The filter criteria is applied on the standard field. Available in API version 63.0 and later.
`Value`	**Type** string **Properties** Filter, Group, Sort **Description** Required. The value to be used in the filter criteria.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

BillingBatchFilterCriteriaHistory on page 1680
History is available for tracked fields of the object.

BillingMilestonePlan

Represents a structured approach to invoicing where invoices are scheduled based on predefined milestones. This object is available in API version 63.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

You need Revenue Cloud Billing and the Billing Admin permission set to access this object.

Fields

Field	Details
`BillingTreatmentId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Required. The billing treatment associated with the billing milestone plan. This field is a relationship field. **Relationship Name** BillingTreatment **Refers To** BillingTreatment
`Description`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Additional details about the billing milestone plan.

Field	Details
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a billing milestone plan record indirectly, for example, through a list view or related record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a billing milestone plan record. If this value is null, it's possible that the user only accessed the billing milestone plan record or a related list view (`LastReferencedDate`), but not viewed the billing milestone plan record itself.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** Required. The name of the billing milestone plan.
`ReferenceItemAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total reference item amount to be billed.
`ReferenceItemId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** Required. The ID of the reference item that links a billing milestone plan item to the original order item. This field is a polymorphic relationship field. **Relationship Name** ReferenceItem

Field	Details
	Refers To BillingSchedule, OrderItem
`Status`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. The status of the billing milestone plan. Valid values are: • `Active` • `Cancelled` • `Completely Billed` • `Draft`

BillingMilestonePlanItem

Represents a specific billing milestone within the billing milestone plan that's used to manage and track billing based on the completion of certain deliverables or stages. This object is available in API version 63.0 and later.

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.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

You need Revenue Cloud Billing and the Billing Admin permission set to access this object.

Fields

Field	Details
`BillingMilestonePlanId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** The billing milestone plan associated with the billing milestone plan item.

Field	Details
	This field is a relationship field. **Relationship Name** BillingMilestonePlan **Relationship Type** Master-detail **Refers To** BillingMilestonePlan (the master object)
`CommencementDate`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The action that triggers the start of the billing milestone plan for a date-based milestone. Valid value is: • `OrderProductActivation`
`CommencementDateOffset`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The offset applied to the commencement date to determine the milestone achievement date for a date-based milestone.
`CommencementDateOffsetUnit`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The unit of time for the commencement date offset. Valid values are: • `Days` • `Months` • `Years`
`Description`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description Additional details about the billing milestone plan item.
`FlatAmount`	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The amount in terms of units of currency, such as $10 or $21.52, to invoice from the order item. Used only when `Type` field has a value of `FlatAmount`.
`IsMilestoneAccomplished`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. Indicates whether the billing treatment is for milestone billing (`true`) or not (`false`).
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a billing milestone plan item record indirectly, for example, through a list view or related record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a billing milestone plan item record. If this value is null, it's possible that the user only accessed the billing milestone plan item record or a related list view (`LastReferencedDate`), but not viewed the billing milestone plan item record itself.
`MilestoneAccomplishmentDate`	**Type** date **Properties** Filter, Group, Nillable, Sort

Field	Details
	Description The date the milestone is achieved for date-based milestones. For event-based milestones, this field indicates the date when the milestone is manually marked as completed.
`MilestoneAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount that's billed when the milestone is reached or completed.
`MilestoneType`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The milestone type for the billing treatment item. Valid values are: • `Date` • `Event`
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** Required. The name of the billing milestone plan item.
`Percentage`	**Type** percent **Properties** Create, Filter, Nillable, Sort, Update **Description** The percentage,such as 10% or 12.5%, to invoice from the order item. Used only when `Type` field has a value of `Percentage`.
`ServicePeriodEnd`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field	Details
	Description The end date of the service associated with the milestone. Valid value is: • `Order Product End Date`
`ServicePeriodStart`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The start date of the service associated with the milestone. Valid value is: • `Order Product Start Date`
`Status`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The status of the billing milestone plan item. Valid values are: • `Cancelled` • `Draft` • `Error` • `Invoiced` • `Ready for Invoicing` • `Waiting for Milestone Accomplishment`
`Type`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. Specifies whether billing schedules created from this billing treatment item are based on a flat amount or a percentage of the order item's total amount. Valid values are: • `FlatAmount` • `Percentage` • `Remainder`

BillingPeriodItem

Represents a payment period for an invoice. The billing period item is used to pass billing information to an invoice line. This object is available in API version 62.0 and later.

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.

Supported Calls

`describeLayout(),describeSObjects(),getDeleted(),getUpdated(),query(),retrieve(),search()`

Special Access Rules

You need the Billing Admin permission set to access this object.

Fields

Field	Details
`Amount`	**Type** currency **Properties** Filter, Sort **Description** Required. The price for the billing period item. This field is used to calculate the invoice line's amount.
`BillingPeriodEndDate`	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The end date of a billing period that's used to calculate the invoice line's end date.
`BillingPeriodItemNumber`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. An auto-generated number identifying the billing period item.
`BillingPeriodStartDate`	**Type** date

Field	Details
	Properties Filter, Group, Nillable, Sort **Description** The start date of a billing period item that's used to calculate the invoice line's start date.
`BillingScheduleId`	**Type** reference **Properties** Filter, Group, Sort **Description** Required. The ID of the parent billing schedule record of the billing period item. This field is a relationship field. **Relationship Name** BillingSchedule **Relationship Type** Master-detail **Refers To** BillingSchedule (the master object)
`InvoiceBatchRunId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of the parent invoice batch run record of the billing period item. This field is a relationship field. **Relationship Name** InvoiceBatchRun **Refers To** InvoiceBatchRun
`InvoiceLineId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of the invoice line record that's generated from the billing period item. This field is populated only when a billing period item is generated by an invoice batch run. Otherwise, this field is empty. This field is a relationship field.

Field	Details
	Relationship Name InvoiceLine **Refers To** InvoiceLine
`InvoiceStatus`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Required. The status of the invoice that contains the invoice line created from the billing period item. Valid values are: • `Canceled`— The invoice for this billing period item was canceled. • `Draft`— The invoice has been created but hasn't been posted. • `DraftInProgress`— The invoice hasn't been created yet. When the invoice is created, the `InvoiceStatus` field value is changed to `Draft`. If the invoice generation process fails, the `InvoiceStatus` field value shows `DraftInProgress`. • `Error`— The invoice for this billing period item was generated in error. • `Pending`— The invoice for this billing period item is being generated. • `Posted`— An invoice line based on this billing period has been created and added successfully to the invoice. • `PostingInProgress`— An invoice line based on this billing period has been created and is in the process of being added to the invoice. • `Voided`— An invoice line based on this billing period was voided. • `VoidInProgress`— An invoice line based on this billing period is in the process of being voided.
`OverageQuantity`	**Type** double **Properties** Filter, Nillable, Sort **Description** The quantity of the usage overage that has exceeded the initial grant. This field is available in API version 63.0 and later with Revenue Cloud Billing.
`RelatedMilestonePlanItemId`	**Type** reference **Properties** Filter, Group, Nillable, Sort

Field	Details
	Description The billing milestone plan item that creates the billing period item. This field is available in API version 63.0 and later with Revenue Cloud Billing. This field is a relationship field. **Relationship Name** RelatedMilestonePlanItem **Refers To** BillingMilestonePlanItem
`RelatedUsageBillPeriodItemId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The method for retrieving billing information for usage-based products. This field is available in API version 63.0 and later with Revenue Cloud Billing. This field is a relationship field. **Relationship Name** RelatedUsageBillPeriodItem **Refers To** UsageBillingPeriodItem
`Status`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Required. The status of the billing period item. Draft billing period items aren't evaluated for invoice line creation. Valid values are: • `Canceled` • `Draft` • `Reviewed`
`TotalUsedQuantity`	**Type** double **Properties** Filter, Nillable, Sort

Field	Details
	Description The total quantity of the usage resource consumed during the billing period. This includes granted entitlements, excess consumption, or consumption below the granted entitlements. This field is available in API version 63.0 and later with Revenue Cloud Billing.
`UnitOfMeasureId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The units used to measure the overage. For example, bytes or minutes. Available in API version 63.0 and later. This field is a relationship field. **Relationship Name** UnitOfMeasure **Refers To** UnitOfMeasure

BillingPolicy

Represents information about a set of billing treatments that define the rules to invoice a customer for an order item. This object is available in API version 62.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

You need the Billing Admin permission set to access this object.

Fields

Field	Details
`BillingTreatmentSelection`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update

Field	Details
	Description Required. Defines how billing treatments are assigned to order items and assets that are related to the billing policy. Valid values are: • `Default` • `LegalEntity` • `Manual`
`DefaultBillingTreatmentId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** When `BillingTreatmentSelection` has a value of `Default`, the selected billing treatment is used for all the order items and assets that are related to the billing policy. This field is a relationship field. **Relationship Name** DefaultBillingTreatment **Refers To** BillingTreatment
`Description`	**Type** textarea **Properties** Create, Filter, Nillable, Sort, Update **Description** Additional details about the billing policy.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a billing policy indirectly, for example, through a list view or related record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort

Field	Details
	Description The timestamp when the current user last viewed a billing policy. If this value is null, it's possible that the user only accessed the billing policy or a related list view (`LastReferencedDate`), but not viewed the accounting period itself.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** Required. The name of the billing policy.
`Status`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. The status of the billing policy. Valid values are: • `Active` • `Draft` • `Inactive`

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

BillingPolicyHistory on page 1680
History is available for tracked fields of the object.

BillingSchedule

Represents information about the order item that's used in the invoicing process. This object is available in API version 62.0 and later.

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.

Supported Calls

`describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), update()`

Special Access Rules

You need the Billing Admin permission set to access this object.

Fields

Field	Details
`BillDayOfMonth`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The day of the month on which a recurring billing process is scheduled to occur.
`BilledAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total amount (excluding tax) that has been invoiced from the billing schedule. This field is a calculated field.
`BillingAccountId`	**Type** reference **Properties** Filter, Group, Sort **Description** Required. The ID of the account that's related to the billing schedule. This field is a relationship field. **Relationship Name** BillingAccount **Refers To** Account
`BillingMethod`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** The type of billing used for the source item. Valid values are: • `Evergreen`

Field	Details
	• `OrderAmount`
`BillingMilestonePlanId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The billing milestone plan associated with the billing schedule. This field is available in API version 63.0 and later with Revenue Cloud Billing. This field is a relationship field. **Relationship Name** BillingMilestonePlan **Refers To** BillingMilestonePlan
`BillingPeriodAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount to be invoiced each billing period. For example, if the billing period is monthly, this field shows the monthly amount that appears on the invoice line.
`BillingScheduleEndDate`	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The last date that the billing schedule is available for invoicing. This value is inherited from the `EndDate` field on the order item.
`BillingScheduleGroupId`	**Type** reference **Properties** Filter, Group, Sort **Description** Required. The billing schedule group that contains the billing schedule. Billing schedules are grouped when they have the same source order item. The source order item is the original order item that a customer bought. Afterwards, if the customer amends, cancels, or renews the order item, a new billing schedule is created with the `BillingScheduleGroupId` for the original order item. This field is a relationship field.

Field	Details
	Relationship Name BillingScheduleGroup **Relationship Type** Master-detail **Refers To** BillingScheduleGroup (the master object)
`BillingScheduleNumber`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. The auto-generated reference number for the billing schedule.
`BillingScheduleStartDate`	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The date when the billing schedule is available for invoicing. This value is inherited from the `ServiceDate` field on the order item.
`BillingTerm`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The duration for which the customer is invoiced.
`BillingTermUnit`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Required. The unit of measurement of the billing term. Valid values are: • `BillingMilestonePlan`—Available in API version 63.0 and later with Revenue Cloud Billing. • `Day` • `Month` • `OneTime`

Field	Details
	• `Quarter` • `Semi-Annual` • `Year`
`BillingTreatmentItemId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The billing treatment item that's used to configure invoiceable amounts on the billing schedule. This field is a relationship field. **Relationship Name** BillingTreatmentItem **Refers To** BillingTreatmentItem
`BillingType`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The timing of invoicing for a product or service relative to its delivery to the customer. Valid values are: • `Advance`— Invoices a product or service before its delivery. • `Arrears` — Invoices a product or service after its delivery.
`CancellationDate`	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The date from when the user can no longer access the service. For example, if a service ends on August 31, the cancellation date is September 1 because that's the date from when the user can no longer use the service. Billing schedules past their cancellation date aren't invoiced.
`Category`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort

Field	Details
	Description Required. The category of the billing schedule. Valid values are: • `AmendQuantity`— A billing schedule for an order that changes the quantity. • `Cancellation` — A billing schedule for an order that was canceled. • `Original` — A billing schedule for the initial order. • `Renewal` — A billing schedule for an order that was renewed.
`CustomInvoiceGroupKey`	**Type** string **Properties** Filter, Group, Restricted picklist, Sort **Description** The group identifier for which an invoice must be generated when the invoice group type is `Custom`. This field is available in API version 63.0 and later with Revenue Cloud Billing.
`InvBatchDraftToPostedRunId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The batch job that posts all invoices with a status as `Draft` that are generated by the invoice batch run associated with the billing schedule. This field is a relationship field. **Relationship Name** InvBatchDraftToPostedRun **Refers To** InvBatchDraftToPostedRun
`InvoiceBatchRunId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The invoice batch run that evaluated the billing schedule and its billing period items to generate an invoice. This field is a relationship field. **Relationship Name** InvoiceBatchRun

Field	Details
	Refers To InvoiceBatchRun
`InvoiceGroupType`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Specifies whether to generate an invoice for a billing schedule, a custom group, or the default group of account, bill-to-contact, payment term, currency, and tax engine. Valid values are: • `Billing Schedule` • `Custom` • `Default` This field is available in API version 63.0 and later with Revenue Cloud Billing.
`InvoiceRunMatchingValue`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The batch value used by the invoice run that evaluated the billing schedule. During an invoice run, billing schedules with the same batch value, including null, are grouped to the same invoice run.
`LineAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount associated with a specific line item in the billing schedule. Available in API version 63.0 and later.
`NextBillingDate`	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The date that the next billing period starts for the invoice. This date is used to calculate which invoice lines are included on an invoice. When an invoice scheduler or API evaluates an order for invoicing, billing schedules with a next billing date on or before the invoice's target date are included in the invoice.

Field	Details
`NextChargeFromDate`	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The date that the billing schedule is invoiced in the upcoming billing period. For example, if you invoiced a customer for a billing period of 01/01/24 through 01/31/24, the billing schedule's `NextChargeFromDate` is 02/01/22.
`OriginalBillingScheduleId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** If the billing schedule is an amended or canceled billing schedule, then this field shows the original billing schedule. Otherwise, this field is null. This field is a relationship field. **Relationship Name** OriginalBillingSchedule **Refers To** BillingSchedule
`PendingAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount from the current billing term that hasn't been billed yet. For example, the unbilled amount for a month, quarter, or year, depending on this billing schedule's billing term.
`Quantity`	**Type** double **Properties** Filter, Nillable, Sort **Description** The quantity of the order item that created the billing schedule.
`ReferenceEntityId`	**Type** reference **Properties** Filter, Group, Nillable, Sort

Field	Details
	Description The parent order of the order item that created the billing schedule. This field is a polymorphic relationship field. **Relationship Name** ReferenceEntity **Refers To** Order
`ReferenceEntityItemId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The order item or asset that created the billing schedule. This field is a polymorphic relationship field. **Relationship Name** ReferenceEntityItem **Refers To** OrderItem, OrderItemAdjustmentLineItem
`Status`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. The status of the order item that the billing schedule represents. Valid values are: • `WaitingForMilestoneEventAccomplishment`—Available in API version 63.0 and later with Revenue Cloud Billing. • `CompletelyBilled` • `Error` • `Processing` • `ReadyForInvoicing`
`TaxTreatmentId`	**Type** reference **Properties** Filter, Group, Sort

Field	Details
	Description Required. The ID of the treatment that's used to calculate tax for the billing schedule. This value is defined based on the order item's tax policy. This field is a relationship field. **Relationship Name** TaxTreatment **Refers To** TaxTreatment
`TotalAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total amount of the order item represented by the billing schedule.
`UnitOfMeasureId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of the unit of measure for the billing schedule. Available in API version 63.0 and later. This field is a relationship field. **Relationship Name** UnitOfMeasure **Refers To** UnitOfMeasure
`UnitPrice`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The price for an individual unit of the billing schedule's parent order item, including charges, adjustments, and discounts. This value is inherited from the order item's `UnitPrice` field.
`UsageResourceId`	**Type** reference **Properties** Filter, Group, Nillable, Sort

Field	Details
	Description The usage resource associated with the billing schedule. This field is available in API version 63.0 and later with Revenue Cloud Billing. This field is a relationship field. **Relationship Name** UsageResource **Refers To** UsageResource

BillingScheduleGroup

Represents a consolidated view of all the billing schedules related to the order items generated from one asset, including new orders and amendment orders. This object is available in API version 62.0 and later.

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.

Supported Calls

`delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `update()`

Special Access Rules

You need the Billing Admin permission set to access this object.

Fields

Field	Details
`AnchorProdtBillingSchdGrpId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The billing schedule group for the anchor product in a usage-based product. The anchor product contains the grants for the usage-based product. This field is available in API version 63.0 and later with Revenue Cloud Billing. This field is a relationship field. **Relationship Name** AnchorProdtBillingSchdGrp

Field	Details
	Refers To BillingScheduleGroup
`BillDayOfMonth`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The day of the month on which a recurring billing process is scheduled to occur.
`BillToContactId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ID of the contact related to the billing schedule group. This field can't be modified when related billing schedules are in processing. This field is a relationship field. **Relationship Name** BillToContact **Refers To** Contact
`BillingAccountId`	**Type** reference **Properties** Filter, Group, Sort **Description** Required. The ID of the account that's related to the billing schedule group. This field is a relationship field. **Relationship Name** BillingAccount **Refers To** Account
`BillingAddress`	**Type** address **Properties** Filter, Nillable

Field	Details
	Description The compound form of the billing address. Read only. See Address Compound Fields for details on compound address fields.
`BillingMethod`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Required. The type of billing used for the source item. Valid values are: • `Evergreen` • `OrderAmount` • `Usage`
`BillingResumptionDate`	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The date when billing for the asset related to the billing schedule group is resumed. Available in API version 63.0 and later.
`BillingScheduleGroupNumber`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. The auto-generated reference number for the billing schedule group.
`BillingStartMonth`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** This is a read-only field used with annual billing. The field shows the numbers from 1 through 12, which indicate the month when billing begins for an annual subscription. For example, if billing starts in January, the value is 1; if billing starts in June, the value is 6.
`BillingSuspensionDate`	**Type** date

Field	Details
	Properties Filter, Group, Nillable, Sort **Description** The date when billing for the asset related to the billing schedule group is suspended. Available in API version 63.0 and later.
`BillingTerm`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** This field is used with `BillingTermUnit`field to define a billing cycle. For example, bill every 20 days or every two months. In this example, the `BillingTerm` field value is 20 and the `BillingTermUnit` field value is `Day`.
`BillingTermUnit`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Required. The unit of measurement of the billing term. Valid values are: • `BillingMilestonePlan`—Available in API version 63.0 and later with Revenue Cloud Billing. • `Day` • `Month` • `OneTime` • `Quarter` • `Semi-Annual` • `Year`
`BillingTreatmentId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The billing treatment item that's used to configure invoiceable amounts on the billing schedule group. This field is a relationship field. **Relationship Name** BillingTreatment

Field	Details
	Refers To BillingTreatment
`BillingType`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Required. The timing of invoicing for a product or service relative to its delivery to the customer. Valid values are: • `Advance`— Invoices a product or service before its delivery. • `Arrears`— Invoices a product or service after its delivery.
`BindingInstanceId`	**Type** reference **Properties** Filter, Group, Nillable, Sort, Update **Description** The asset or custom object record associated with the billing schedule group. This field is available in API version 63.0 and later with Revenue Cloud Billing. This field is a polymorphic relationship field. **Relationship Name** BindingInstance **Refers To** Asset
`CancellationDate`	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The date from when the user can no longer access the service. For example, if a service ends on August 31, the cancellation date is September 1 because that's the date from when the user can no longer use the service. Billing schedules past their cancellation date aren't invoiced.
`Controller`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort

Field	Details
	Description Required. During the invoicing process, this field determines which date is used when the billing schedule group and billing schedule have a related field with conflicting values. For example, when Controller has a value of `BillingScheduleGroup`, if the billing schedule's billing day of month is 5 while the billing schedule group's billing day of month is 10, the invoice is sent on the 10th day of the month. Valid values are: • `BillingScheduleGroup` • `None`
`EffectiveNextBillingDate`	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The earliest `NextBillingDate` from all billing schedules in the billing schedule group. This field is a reference field that isn't used for any features or calculations. This field is a calculated field.
`EndDate`	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The end date of the billing schedule group.
`ExternalBindingInstance`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The custom target associated with the entitlements that are granted with the sellable product. This field is available in API version 63.0 and later with Revenue Cloud Billing.
`LegalEntityId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of the legal entity that's related to the billing schedule group. This field is a relationship field.

Field	Details
	Relationship Name LegalEntity **Refers To** LegalEntity
NextBillingDateOverride	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The next billing date for all the billing schedules related to the billing schedule group. If specified, this date overrides the next billing dates of the billing schedules. Available in API version 63.0 and later.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. The ID of the user who created the billing schedule group. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
PaymentTermId	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ID of the payment term used in this billing schedule group. This field can't be modified when related billing schedules are in processing. This field is a relationship field. **Relationship Name** PaymentTerm **Refers To** PaymentTerm
PeriodBoundary	**Type** picklist

Field	Details
	Properties Filter, Group, Nillable, Restricted picklist, Sort **Description** This field is inherited from the order item's parent quote line item or sales transaction item. The period boundary determines the start and end date of the billing periods. Valid values are: • `AlignToCalendar` • `Anniversary` • `DayOfPeriod` • `LastDayOfPeriod`
`Product2Id`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The product that was charged or ordered to create the billing schedule group. This field is a relationship field. **Relationship Name** Product2 **Refers To** Product2
`ProductName`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The name of the product for the order item that's represented by each billing schedule in the billing schedule group.
`ProrationPolicyId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of the proration policy that applies to this billing schedule group. The proration policy defines how time periods are calculated for orders. For example, whether partial periods are allowed. This field is inherited from the shared proration policy for each billing schedule in the billing schedule group. This field is a relationship field.

Field	Details
	Relationship Name ProrationPolicy **Refers To** ProrationPolicy
`ReferenceEntityId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The asset used to create the billing schedules in the billing schedule group. This field is a relationship field. **Relationship Name** ReferenceEntity **Refers To** Asset
`ShippingAddress`	**Type** address **Properties** Filter, Nillable **Description** The compound form of the shipping address. See Address Compound Fields for details on compound address fields. This field is a read-only field.
`StartDate`	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The earliest start date from all billing schedules in the billing schedule group.
`TaxTreatmentId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of the tax treatment record that's used to calculate tax for the billing schedule group. This value is defined based on the order item's tax policy. This field is a relationship field.

Field	Details
	Relationship Name TaxTreatment **Refers To** TaxTreatment
`TotalBilledAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount that has been invoiced for the billing schedules within the billing schedule group. This field is a calculated field.
`TotalPendingAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount that hasn't yet been invoiced for the billing schedules within the billing schedule group. This field is a calculated field.

BillingTreatment

Represents information about the billing of an order item. This object is available in API version 62.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

You need the Billing Admin permission set to access this object.

Fields

Field	Details
`BillingPolicyId`	**Type** reference

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update **Description** The billing policy that's related to the billing treatment. This field is a relationship field. **Relationship Name** BillingPolicy **Refers To** BillingPolicy
`Description`	**Type** textarea **Properties** Create, Filter, Nillable, Sort, Update **Description** Additional details about the billing treatment.
`ExcludeFromBilling`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. Specifies whether any order items assigned to the treatment are excluded from billing. Valid values are: • `No` • `Yes`
`IsMilestoneBilling`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. Indicates whether the billing treatment is for milestone billing (`true`) or not (`false`). The default value is `false`. This field is available in API version 63.0 and later with Revenue Cloud Billing.
`LastReferencedDate`	**Type** dateTime

Field	Details
	Properties Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a billing treatment indirectly, for example, through a list view or related record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a billing treatment. If this value is null, it's possible that the user only accessed the billing treatment or a related list view (`LastReferencedDate`), but not viewed the billing treatment itself.
`LegalEntityId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The legal entity that's used to assign the treatment to order items when the parent billing policy's `BillingTreatmentSelection` field value is `LegalEntity`. This field is a relationship field. **Relationship Name** LegalEntity **Refers To** LegalEntity
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** Required. The name of the billing treatment.
`Status`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. The status of the billing treatment.

Field	Details
	Valid values are: • `Active` • `Draft` • `Inactive` Draft or inactive billing treatments can't be assigned to order items.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

BillingTreatmentHistory on page 1680
History is available for tracked fields of the object.

BillingTreatmentItem

Represents information about allocation of the total amount of an order item to billing schedules throughout the order item's lifecycle. This object is available in API version 62.0 and later.

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.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

You need the Billing Admin permission set to access this object.

Fields

Field	Details
`BillingTreatmentId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Required. The parent billing treatment for the billing treatment item. This field is a relationship field.

Field	Details
	Relationship Name BillingTreatment **Relationship Type** Master-detail **Refers To** BillingTreatment (the master object)
`BillingType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. The timing of invoicing for a product or service relative to its delivery to the customer. Advance billing invoices a product or service before its delivery, whereas arrears billing invoices it after delivery. The billing system assesses the billing type to determine the next billing date for an order product. Valid values are: • `Advance`—If the order item is billed in advance, the order's billing day of month is evaluated to choose the nearest date on or before the order product's start date. For example, if a monthly order product's start date is January 1, and the order's billing day of month is 15, the next billing date is December 15. • `Arrears`—If the order item is billed in arrears, the order's billing day of month is evaluated to choose the nearest date after the order product's start date. For example, if a monthly order product's start date is January 1, and the order's billing day of month is 15, the order product's next billing date is January 15. • `None`
`Controller`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. During the invoicing process, this field determines which value is used when the billing schedule group and billing schedule have a shared field with different values. For example, when Controller has a value of `BillingScheduleGroup`, if the billing schedule's billing day of month is 5 while the billing schedule group's billing day of month is 10, the value of 10 is used. Valid values are: • `BillingScheduleGroup` • `None`
`Description`	**Type** textarea

Field	Details
	Properties Create, Filter, Nillable, Sort, Update **Description** Additional details about the billing treatment item.
`FlatAmount`	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The amount in terms of units of currency (such as $10 or $21.52) to invoice from the order item. Used only when Type has a value of `FlatAmount`.
`Handling0Amount`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. Specifies how invoices are generated for billing period items that have an amount of $0. Valid values are: • `CreateInvoice` • `None`
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a billing treatment item indirectly, for example, through a list view or related record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a billing treatment item. If this value is null, it's possible that the user only accessed the billing treatment item or a related list view (`LastReferencedDate`), but not viewed the billing treatment item itself.

Field	Details
`MilestoneStartDate`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The occurrence whose date is used to calculate the milestone accomplishment date for the associated billing milestone plan item based on the provided offset details. Valid value is: • `OrderProductActivation` This field is available in API version 63.0 and later with Revenue Cloud Billing.
`MilestoneStartDateOffset`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The calculated value that determines the time difference from the start of a milestone. This value is calculated from the milestone start date record. This field is available in API version 63.0 and later with Revenue Cloud Billing.
`MilestoneStartDateOffsetUnit`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The unit type to use with the milestone start date offset, which defines the milestone date. Valid values are: • `Days` • `Months` • `Years` This field is available in API version 63.0 and later with Revenue Cloud Billing.
`MilestoneType`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The milestone type for the billing treatment item. Valid values are: • `Date`

Field	Details
	• `Event` This field is available in API version 63.0 and later with Revenue Cloud Billing.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** Required. The name of the billing treatment item.
`Percentage`	**Type** percent **Properties** Create, Filter, Nillable, Sort, Update **Description** The percentage (such as 10% or 12.5%) to invoice from the order item. Used only when `Type` field has a value of `Percentage`.
`ProcessingOrder`	**Type** int **Properties** Create, Filter, Group, Sort, Update **Description** Required. The order in which billing schedules are created based on each billing treatment item. Lower numbers are evaluated first. For example, if your billing treatment has a billing treatment item that invoices at 25 percentage and a `ProcessingOrder` of 1, and another item that invoices at 75 percentage and a `ProcessingOrder` of 2, your first billing schedule will be for 25% of the order item's total amount, and your second billing schedule will be for 75% of the order item's total amount.
`Sequencing`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. The number used to start invoice numbers on invoices generated from this billing treatment item. Valid values are: • `Manual` • `None`
`Status`	**Type** picklist

Field	Details
	Properties Create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. The status of the billing treatment item. Valid values are: • `Active` • `Draft`– Draft billing treatment items aren't evaluated for creating billing schedules.
`Type`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. Specifies whether billing schedules created from this billing treatment item are based on a flat amount or a percentage of the order item's total amount. Valid values are: • `FlatAmount` • `Percentage` • `Remainder`

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

BillingTreatmentItemHistory on page 1680
History is available for tracked fields of the object.

BsgRelationship

Represents a relationship between billing schedule groups to support bundles where one parent billing schedule group has multiple child billing schedule groups. This object is available in API version 62.0 and later.

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.

Supported Calls

`describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

You need the Billing Admin permission set to access this object.

Fields

Field	Details
`AssociatedBsgId`	**Type** reference **Properties** Filter, Group, Sort **Description** Required. The ID of the related billing schedule group. In a bundle relationship, this billing schedule group is the child. This field is a relationship field. **Relationship Name** AssociatedBsg **Refers To** BillingScheduleGroup
`AssociatedBsgPricing`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Required. This field describes how the related billing schedule group is priced relative to the primary billing schedule group. Valid values are: • `IncludedInBundlePrice` • `NotIncludedInBundlePrice`
`AssociatedBsgRole`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Required. This field describes the role of the related billing schedule group in the relationship. Valid values are: • `AddOnComponent` • `BundleComponent` • `ClassificationComponent` • `SetComponent`
`LastReferencedDate`	**Type** dateTime

Field	Details
	Properties Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a billing schedule group record indirectly, for example, through a list view or related record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a billing schedule group relationship record. If this value is null, it's possible that the user only accessed the billing schedule group relationship record or a related list view (`LastReferencedDate`), but not viewed the billing schedule group relationship record itself.
`MainBsgId`	**Type** reference **Properties** Filter, Group, Sort **Description** Required. The ID of the primary billing schedule group. In a bundle relationship, this billing schedule group is the parent. This field is a relationship field. **Relationship Name** MainBsg **Relationship Type** Master-detail **Refers To** BillingScheduleGroup (the master object)
`MainBsgRole`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Required. This field describes the role of the primary billing schedule group in the relationship. Valid values are: • `AddOn` • `Bundle` • `Set`

Field	Details
Name	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. The auto-generated reference number for the billing schedule relationship.
ProductRelationshipTypeId	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of the relationship type between the main and associated billing schedule group. This field is a relationship field. **Relationship Name** ProductRelationshipType **Refers To** ProductRelationshipType

CreditMemo

Represents a document that's used to reduce the amount that a buyer owes a seller under the terms of an earlier invoice. This object is available in API version 62.0 and later.

Supported Calls

`describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `update()`

Special Access Rules

You need the Credit Memo Operations User permission set to access this object.

Fields

Field	Details
Balance	**Type** currency **Properties** Filter, Nillable, Sort

Field	Details
	Description The amount of the credit memo that's available for allocation.
`BillToContactId`	**Type** reference **Properties** Filter, Group, Nillable, Sort, Update **Description** This field is inherited from the account's `Bill to Account` field. This field is a relationship field. **Relationship Name** BillToContact **Refers To** Contact
`BillingAccountId`	**Type** reference **Properties** Filter, Group, Sort, Update **Description** Required. The ID of the customer account associated with this credit memo. This field is a relationship field. **Relationship Name** BillingAccount **Refers To** Account
`CorpCrcyCnvTotAmtWithTax`	**Type** double **Properties** Filter, Nillable, Sort **Description** The sum of the total amount with tax on the credit memo in corporate currency. Available in API version 63.0 and later.
`CorporateCurrencyCvsnDate`	**Type** date **Properties** Filter, Group, Nillable, Sort

Field	Details
	Description The date on which the credit memo amounts are converted to the corporate currency. Available in API version 63.0 and later.
`CorporateCurrencyCvsnRate`	**Type** double **Properties** Filter, Nillable, Sort **Description** The exchange rate that's used to convert credit memo amounts to the corporate currency. Available in API version 63.0 and later.
`CorporateCurrencyIsoCode`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The currency ISO code of the corporate currency. Available in API version 63.0 and later.
`CreationMode`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** Specifies whether the credit memo originated in Salesforce or an external system. Valid values are: • `External` • `Salesforce`
`CreditDate`	**Type** date **Properties** Filter, Group, Sort, Update **Description** Required. The date when the credit memo was posted.
`CreditMemoNumber`	**Type** string **Properties** Filter, Group, Nillable, Sort, Update

Field	Details
	Description The number of the credit memo.
`Description`	**Type** string **Properties** Filter, Group, Nillable, Sort, Update **Description** Additional details about the credit memo.
`DocumentNumber`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. The auto-generated reference number for the credit memo.
`EffectiveDate`	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The effective date of the credit memo. If this field is empty, the credit date is used. For reporting purposes only; this field drives no other logic.
`ExternalReference`	**Type** string **Properties** Filter, Group, Nillable, Sort, Update **Description** An external system's ID for the credit memo.
`ExternalReferenceDataSource`	**Type** string **Properties** Filter, Group, Nillable, Sort, Update **Description** The name of the external system that also contains the credit memo.
`LastReferencedDate`	**Type** dateTime

Field	Details
	Properties Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a credit memo record indirectly, for example, through a list view or related record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a credit memo record. If this value is null, it's possible that the user only accessed the credit memo record or a related list view (`LastReferencedDate`), but not viewed the credit memo record itself.
`NetCreditsApplied`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total difference between the credit applied to and credit unapplied from the invoice. This field is a calculated field.
`OwnerId`	**Type** reference **Properties** Filter, Group, Sort, Update **Description** Required. The ID of the user who owns a credit memo record. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
`ReferenceEntityId`	**Type** reference **Properties** Filter, Group, Nillable, Sort, Update

Field	Details
	Description The ID of the record that this credit memo was generated from. For example, the order, order summary, or invoice. This field is a polymorphic relationship field. **Relationship Name** ReferenceEntity **Refers To** Invoice, Order
`SettlementLevel`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** Specifies whether the credit memo amount was applied to an invoice or an invoice line.
`SourceAction`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** Specifies which Salesforce API created the credit memo. Valid values are: • `Invoice` • `NegativeInvoiceLineConversion` • `Standalone` • `VoidPostedInvoice`
`Status`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort, Update **Description** Required. The status of the credit memo. Valid values are: • `Canceled` • `Draft` • `Error` • `Pending` • `Posted`

Field	Details
	• `Voided`
`TotalAdjustmentAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of the `TotalAmount` field values for the credit memo's adjustment lines. This field is a calculated field.
`TotalAdjustmentAmountWithTax`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of the credit memo's adjustment line amounts, including tax.
`TotalAdjustmentTaxAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of the credit memo's adjustment line tax. Adjustment line balances are excluded.
`TotalAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of the credit memo's `TotalLineAmount` and `TotalAdjustmentAmount` field values. This field is a calculated field.
`TotalAmountWithTax`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total credit memo amount, with tax included. This field is a calculated field.

Field	Details
`TotalChargeAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of the `TotalAmount` field values for the credit memo's charge lines. This field is a calculated field.
`TotalChargeAmountWithTax`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of the credit memo's charge line amounts, including tax.
`TotalChargeTaxAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of the credit memo's charge tax amount.
`TotalCreditAmountApplied`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The credit memo amount that's been applied to invoices.
`TotalCreditAmountUnapplied`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The credit memo amount that's been unapplied from invoices.
`TotalTaxAmount`	**Type** currency **Properties** Filter, Nillable, Sort

Field	Details
	Description The sum of the `TotalAmount` field values for the credit memo's tax lines. This field is a calculated field.

CreditMemoAddressGroup

Represents the storage of the buyer's address information, which is used to determine the tax credit amount for a buyer when a credit memo is issued. This object is available in API version 62.0 and later.

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.

Supported Calls

`delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`

Special Access Rules

You need the Credit Memo Operations User permission set to access this object.

Fields

Field	Details
`Address`	**Type** address **Properties** Filter, Nillable **Description** The buyer's address.
`CreditMemoAddressGroupNumber`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. The auto-generated reference number for the credit memo address group.
`CreditMemoId`	**Type** reference

Field	Details
	Properties Filter, Group, Sort **Description** Required. The ID of the credit memo associated with the address group. This field is a relationship field. **Relationship Name** CreditMemo **Relationship Type** Master-detail **Refers To** CreditMemo (the master object)
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a credit memo address group record indirectly, for example, through a list view or related record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a credit memo address group record. If this value is null, it's possible that the user only accessed the credit memo address group record or a related list view (`LastReferencedDate`), but not viewed the credit memo address group record itself.

CreditMemoInvApplication

Represents information about the application of a credit memo to an invoice. This object is available in API version 62.0 and later.

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.

Supported Calls

`describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `update()`

Special Access Rules

You need the Credit Memo Operations User permission set to access this object.

Fields

Field	Details
`Amount`	**Type** currency **Properties** Filter, Sort **Description** Required. The amount of the credit memo that was applied to or unapplied from the invoice.
`AppliedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date when the credit memo was applied. If the credit memo invoice application's type is `Unapplied`, this value is inherited from the `Applied` date of the credit memo referenced in the `AssociatedLineId` field.
`AssociatedLineId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** ID of the original credit memo invoice application for a credit memo invoice application that represents an unapplied credit memo. This field is a relationship field. **Relationship Name** AssociatedLine **Refers To** CreditMemoInvApplication
`CreditMemoBalance`	**Type** currency **Properties** Filter, Nillable, Sort

Field	Details
	Description The balance of a credit memo after it's applied or unapplied. This field is a snapshot of the credit memo's balance after the action. This field isn't updated after further changes to the credit memo balance.
`CreditMemoId`	**Type** reference **Properties** Filter, Group, Sort **Description** Required. The credit memo that was applied or unapplied. This field is a relationship field. **Relationship Name** CreditMemo **Refers To** CreditMemo
`CreditMemoInvoiceNumber`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. An auto-generated number identifying the credit memo invoice application.
`Date`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date when the credit memo amount was applied to the invoice.
`Description`	**Type** string **Properties** Filter, Group, Nillable, Sort, Update **Description** Additional details about the credit memo applied to an invoice.
`EffectiveDate`	**Type** dateTime

Field	Details
	Properties Filter, Nillable, Sort **Description** The effective date of the application or unapplication of credit. You can provide this value when applying or unapplying the credit memo. This field is optional and provided only for reporting purposes. It doesn't affect the credit memo invoice application's other fields.
`HasBeenUnapplied`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** Specifies whether this credit memo application has been unapplied from the target invoice. Valid values are: • `NA` • `No` • `Yes`
`ImpactAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The net adjustment to the invoice's balance after a credit memo is applied or unapplied. If a credit memo was applied, this value is the negative version of the credit memo invoice application's `Amount` field. If a credit memo was unapplied, this value is the positive version of the credit memo invoice application's `Amount` field. This field is a calculated field.
`InvoiceBalance`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The balance of the credit memo after a credit memo is applied or unapplied. This field is a snapshot of the credit memo's balance after the action. This field isn't updated after further changes to the credit memo balance.
`InvoiceId`	**Type** reference **Properties** Filter, Group, Sort

Field	Details
	Description Required. The invoice to which the credit memo is applied. This field is a relationship field. **Relationship Name** Invoice **Relationship Type** Master-detail **Refers To** Invoice (the master object)
`Type`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Required. Specifies whether the credit memo line application was generated because of an apply action (application) or an unapply action (unapplication). Valid values are: • `Applied` • `Unapplied`
`UnappliedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date when this application was unapplied from the target invoice.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

CreditMemoInvApplicationFeed on page 1673
Feed tracking is available for the object.

CreditMemoInvApplicationHistory on page 1680
History is available for tracked fields of the object.

CreditMemoLine

Represents the product, service, adjustment, or tax line items included in a credit memo. This object is available in API version 62.0 and later.

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.

Supported Calls

`describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), update()`

Special Access Rules

You need the Credit Memo Operations User permission set to access this object.

Fields

Field	Details
`AdjustmentAmount`	**Type** currency **Properties** Filter, Nillable, Sort, Update **Description** The amount of this credit memo line item if its type is Adjustment.
`AdjustmentAmountWithTax`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of the adjustment amount and the adjustment tax amount.
`AdjustmentTaxAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount of the tax related to the adjustment amount.
`Balance`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount of the credit memo line available for allocation.

Field	Details
`BillingAddressId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of the billing address related to this credit memo line. This field is a relationship field. **Relationship Name** BillingAddress **Refers To** CreditMemoAddressGroup
`ChargeAmount`	**Type** currency **Properties** Filter, Nillable, Sort, Update **Description** The amount of this credit memo line item if its type is Charge.
`ChargeAmountWithTax`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of the charge amount and the charge tax amount.
`ChargeTaxAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount of the tax related to the charge amount.
`CorpCurrencyCnvChargeAmt`	**Type** double **Properties** Filter, Nillable, Sort, Update **Description** The amount of the credit memo line item if its type is `Charge` in corporate currency. Available in API version 63.0 and later.

Field	Details
`CorpCurrencyCnvTotalTaxAmt`	**Type** double **Properties** Filter, Nillable, Sort, Update **Description** The sum of the total amount with the tax on the credit memo line in corporate currency. Available in API version 63.0 and later.
`CorporateCurrencyCvsnDate`	**Type** date **Properties** Filter, Group, Nillable, Sort, Update **Description** The date on which the credit memo line amounts are converted to the corporate currency. Available in API version 63.0 and later.
`CorporateCurrencyCvsnRate`	**Type** double **Properties** Filter, Nillable, Sort, Update **Description** The exchange rate that's used to convert the credit memo line amounts to the corporate currency. Available in API version 63.0 and later.
`CorporateCurrencyIsoCode`	**Type** string **Properties** Filter, Group, Nillable, Sort, Update **Description** The currency ISO code of the corporate currency. Available in API version 63.0 and later.
`CreditMemoId`	**Type** reference **Properties** Filter, Group, Sort **Description** Required. The ID of the parent credit memo. This field is a relationship field. **Relationship Name** CreditMemo

Field	Details
	Relationship Type Master-detail **Refers To** CreditMemo (the master object)
`Description`	**Type** string **Properties** Filter, Group, Nillable, Sort, Update **Description** Additional details about the credit memo line.
`EndDate`	**Type** date **Properties** Filter, Group, Nillable, Sort, Update **Description** The description of the credit memo line.
`LegalEntityAccountingPeriodId`	**Type** reference **Properties** Filter, Group, Nillable, Sort, Update **Description** The ID of the legal entity accounting period record of the credit memo line. This field is a relationship field. **Relationship Name** LegalEntityAccountingPeriod **Refers To** LegalEntyAccountingPeriod
`LegalEntityId`	**Type** reference **Properties** Filter, Group, Nillable, Sort, Update **Description** The ID of the legal entity record related to the credit memo line. This field is a relationship field. **Relationship Name** LegalEntity

Field	Details
	Refers To LegalEntity
`LineAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount of the credit memo line. This field is a calculated field.
`Name`	**Type** string **Properties** Filter, Group, idLookup, Sort, Update **Description** Required. The name of the credit memo line.
`NetCreditsApplied`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total amount applied to invoice lines from the credit memo. This amount is calculated by subtracting the total unapplied credit amount from the total applied credit amount. This field is a calculated field.
`Product2Id`	**Type** reference **Properties** Filter, Group, Nillable, Sort, Update **Description** The ID of the product or service being credited in the credit memo line. This field is a relationship field. **Relationship Name** Product2 **Refers To** Product2
`ReferenceEntityItemId`	**Type** reference

Field	Details
	Properties Filter, Group, Nillable, Sort, Update **Description** The order product or invoice line corresponding to this credit memo line. This field is a polymorphic relationship field. **Relationship Name** ReferenceEntityItem **Refers To** InvoiceLine, OrderItem
`ReferenceEntityItemType`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The type of transaction that generated the credit memo line. Valid values are: • `Delivery Charge` • `Fee` • `Order Product`
`ReferenceEntityItemTypeCode`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The type of object that generated the credit memo line. Valid values are: • `Charge` • `Product`
`ShippingAddressId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of the shipping address. This field is a relationship field. **Relationship Name** ShippingAddress

Field	Details
	Refers To CreditMemoAddressGroup
`StartDate`	**Type** date **Properties** Filter, Group, Nillable, Sort, Update **Description** The first date of the billing for the service for credit memo lines generated from a time-based service.
`Status`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The status of the credit memo line. This field is inherited from the credit memo.
`TaxAmount`	**Type** currency **Properties** Filter, Nillable, Sort, Update **Description** The total tax amount for the credit memo.
`TaxTreatmentId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of the tax treatment record for the credit memo line. This field is a relationship field. **Relationship Name** TaxTreatment **Refers To** TaxTreatment

CreditMemoLineInvoiceLine

Represents a junction between a credit memo line and an invoice line. This object is available in API version 62.0 and later.

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.

Supported Calls

`describeLayout(),describeSObjects(),getDeleted(),getUpdated(),query(),retrieve(),update()`

Special Access Rules

You need the Credit Memo Operations User permission set to access this object.

Fields

Field	Details
`Amount`	**Type** currency **Properties** Filter, Sort **Description** Required. The amount that's been applied to or unapplied from the invoice line.
`AppliedDateTime`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date and time when the credit memo line was applied to the invoice line.
`CreditMemoLineBalance`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The balance of the credit memo line after it's applied to or unapplied from the invoice line.
`CreditMemoLineId`	**Type** reference **Properties** Filter, Group, Sort **Description** Required. The ID of the credit memo line record that's applied to or unapplied from the invoice line. This field is a relationship field.

Field	Details
	Relationship Name CreditMemoLine **Refers To** CreditMemoLine
`CreditMemoLineInvoiceLineNumber`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. An auto-generated number identifying the credit memo line invoice line.
`Description`	**Type** textarea **Properties** Filter, Group, Nillable, Sort, Update **Description** Additional details about the credit memo line invoice line.
`EffectiveDateTime`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date and time when the credit memo line's application to or unapplication from the invoice line becomes effective.
`ImpactAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The credit memo line invoice line’s financial impact against the customer’s accounts receivable. If the credit memo line invoice line's type is `Applied`, the impact amount is the negative equivalent of the credit memo line invoice line's amount. Otherwise, the impact amount is equal to the credit memo line invoice line's amount. This field is a calculated field.
`InvoiceLineBalance`	**Type** currency **Properties** Filter, Nillable, Sort

Field	Details
	Description The balance of the invoice line after the credit memo line was applied or unapplied.
`InvoiceLineId`	**Type** reference **Properties** Filter, Group, Sort **Description** Required. The ID of the invoice line record to which the credit memo line has been applied or unapplied. This field is a relationship field. **Relationship Name** InvoiceLine **Relationship Type** Master-detail **Refers To** InvoiceLine (the master object)
`LegalEntityAccountingPeriodId`	**Type** reference **Properties** Filter, Group, Nillable, Sort, Update **Description** The ID of the legal entity accounting period record related to the credit memo line invoice line. This field is a relationship field. **Relationship Name** LegalEntityAccountingPeriod **Refers To** LegalEntyAccountingPeriod
`LegalEntityId`	**Type** reference **Properties** Filter, Group, Nillable, Sort, Update **Description** The ID of the legal entity record related to the credit memo line invoice line. This field is a relationship field. **Relationship Name** LegalEntity

Field	Details
	Refers To LegalEntity
`RelatedCrMemoLineInvcLineId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of the related credit memo line invoice line record of the type `Applied` when the credit memo line invoice line's type is `Unapplied`. This field is a relationship field. **Relationship Name** RelatedCrMemoLineInvcLine **Refers To** CreditMemoLineInvoiceLine
`Type`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Required. Specifies whether the credit memo line has been applied to or unapplied from the invoice line. Valid values are: • `Applied` • `Unapplied`
`UnappliedDateTime`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date and time when the credit memo line was unapplied from the invoice line.
`UnappliedStatus`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Required. Specifies whether the credit memo line has been unapplied from the invoice line. Valid values are:

Field	Details
	• NA • No • Yes

CreditMemoLineTax

Represents tax information of a credit memo line of type `Tax`. This object is available in API version 62.0 and later.

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.

Supported Calls

`describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `update()`

Special Access Rules

You need the Tax Admin permission set to access this object.

Fields

Field	Details
`BillingAddressId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The billing address of the parent credit memo line. This field is a relationship field. **Relationship Name** BillingAddress **Refers To** CreditMemoAddressGroup
`CalculationStatus`	**Type** picklist **Properties** Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort **Description** The status of the tax calculation.

Field	Details
	Valid values are: • `Complete` • `Error` • `None` The default value is `None`.
`CorpCrcyCnvTaxAmount`	**Type** double **Properties** Filter, Nillable, Sort, Update **Description** The total tax amount of the credit memo line tax in corporate currency. Available in API version 63.0 and later.
`CorporateCurrencyCvsnDate`	**Type** date **Properties** Filter, Group, Nillable, Sort, Update **Description** The date on which the credit memo line tax amounts are converted to the corporate currency. Available in API version 63.0 and later.
`CorporateCurrencyCvsnRate`	**Type** double **Properties** Filter, Nillable, Sort, Update **Description** The exchange rate that's used to convert the credit memo line tax amounts to the corporate currency. Available in API version 63.0 and later.
`CorporateCurrencyIsoCode`	**Type** string **Properties** Filter, Group, Nillable, Sort, Update **Description** The currency ISO code of the corporate currency. Available in API version 63.0 and later.
`CreditMemoLineId`	**Type** reference **Properties** Filter, Group, Sort

Field	Details
	Description Required. The ID of the parent charge or adjustment credit memo line record that a credit memo line tax is related to. This field is a relationship field. **Relationship Name** CreditMemoLine **Relationship Type** Master-detail **Refers To** CreditMemoLine (the master object)
`CreditMemoLineTaxNumber`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. An auto-generated number identifying a credit memo line tax.
`Description`	**Type** string **Properties** Filter, Group, Nillable, Sort, Update **Description** Additional details about the credit memo line tax.
`EndDate`	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The end date of a credit memo line tax.
`LegalEntityAccountingPeriodId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of the legal entity accounting period record that's related to a credit memo line tax. This field is a relationship field. **Relationship Name** LegalEntityAccountingPeriod

Field	Details
	Refers To LegalEntyAccountingPeriod
`LegalEntityId`	**Type** reference **Properties** Filter, Group, Nillable, Sort, Update **Description** The ID of the legal entity record that's related to a credit memo line tax. This field is a relationship field. **Relationship Name** LegalEntity **Refers To** LegalEntity
`Product2Id`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The product or service being credited in the parent credit memo line. Available in API version 62.0 only. This field is a relationship field. **Relationship Name** Product2 **Refers To** Product2
`ReferenceEntityItemId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The invoice line tax corresponding to a credit memo line tax. This field is a relationship field. **Relationship Name** ReferenceEntityItem **Refers To** InvoiceLineTax

Field	Details
ShippingAddressId	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The shipping address of the parent credit memo line. This field is a relationship field. **Relationship Name** ShippingAddress **Refers To** CreditMemoAddressGroup
StartDate	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The start date of a credit memo line tax.
TaxAmount	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total amount of a credit memo line tax.
TaxCode	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The code that's used to calculate the tax rate for the parent credit memo line.
TaxDocumentNumber	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The document number that's used to calculate the tax rate for the parent credit memo line.

Field	Details
TaxEffectiveDate	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The date used to calculate the tax amount.
TaxName	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The user-defined name for a credit memo line tax.
TaxRate	**Type** percent **Properties** Filter, Nillable, Sort **Description** The percentage that's used to calculate tax.
TaxTransactionNumber	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The number of the transaction in the external tax engine that calculated tax for the parent credit memo line.
TaxTreatmentId	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of the tax treatment record that's related to a credit memo line tax. This field is a relationship field. **Relationship Name** TaxTreatment **Refers To** TaxTreatment

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

CreditMemoLineTaxFeed on page 1673
Feed tracking is available for the object.

CreditMemoLineTaxHistory on page 1680
History is available for tracked fields of the object.

GeneralLedgerAccount

Represents information about the accounting codes, types, and names that are used to store and organize financial transactions. This object is available in API version 63.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

You need Revenue Cloud Billing and the Accounts Receivables Admin permission set to access this object.

Fields

Field	Details
`AccountingCode`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** The code that's used to organize information about the general ledger account.
`AccountingName`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** The user-specified name for the general ledger account.
`AccountingType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update

Field	Details
	Description The accounting type for the general ledger account.
`Description`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Additional details about the general ledger account.
`FinancialStatement`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The financial statement that's created by using the information from the general ledger account.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed this record indirectly, for example, through a list view or related record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed this record or list view. If this value is null, and `LastReferenceDate` is not null, the user accessed this record or list view indirectly.
`LegalEntityId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The legal entity that's related to the general ledger account. This field is a relationship field.

Field	Details
	Relationship Name LegalEntity **Refers To** LegalEntity
`Name`	**Type** string **Properties** Filter, Group, idLookup, Sort **Description** An auto-generated name identifying the general ledger account, which is a combination of the accounting code and the account name.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The ID of the owner of this object or ID of the creator of this object. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User

GeneralLedgerAcctAsgntRule

Represents information about the rule based on which general ledger accounts are assigned to transaction journals that are created for billing transactions. This object is available in API version 63.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

You need Revenue Cloud Billing and the Accounts Receivables Admin permission set to access this object.

Fields

Field	Details
`CreditGeneralLedgerAccountId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The general ledger account for a credit transaction that's related to the general ledger account assignment rule. This field is a relationship field. **Relationship Name** CreditGeneralLedgerAccount **Refers To** GeneralLedgerAccount
`CurrencyIsoCode`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The currency code of the general ledger account assignment rule for Salesforce orgs with multicurrency enabled. Valid values are: • `AUD`—Australian Dollar • `EUR`—Euro • `SGD`—Singapore Dollar • `USD`—U.S. Dollar The default value is `USD`.
`DebitGeneralLedgerAccountId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The general ledger account for a debit transaction that's related to the general ledger account assignment rule. This field is a relationship field. **Relationship Name** DebitGeneralLedgerAccount

Field	Details
	Refers To GeneralLedgerAccount
`FilterCriteria`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The filter criteria for the general ledger account assignment rule. Valid values are: • `All` • `Any` • `Custom`
`FilterLogic`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The filter logic for the general ledger account assignment rule when the filter criteria is `Custom`. Transaction journals are created for the transactions of the selected type that meet the defined filter logic.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed this record indirectly, for example, through a list view or related record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed this record or list view. If this value is null, and `LastReferenceDate` is not null, the user accessed this record or list view indirectly.
`LegalEntityId`	**Type** reference

Field	Details
	Properties Create, Filter, Group, Nillable, Sort, Update **Description** The legal entity that's related to the general ledger account assignment rule. This field is a relationship field. **Relationship Name** LegalEntity **Refers To** LegalEntity
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** A user-defined name for the general ledger account assignment rule.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** The ID of the owner of the general ledger account assignment rule record. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
`Priority`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The priority of the general ledger account assignment rule when there are multiple general ledger account assignment rules defined for a transaction type.
`Status`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update

Field	Details
	Description The status of the general ledger account assignment rule. Valid values are: • `Active` • `Inactive`
`TransactionType`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The transaction type that's related to the general ledger account assignment rule. Valid values are: • `CreditMemo` • `CreditMemoLine` • `CreditMemoLineTax` • `Invoice` • `InvoiceLine` • `InvoiceLineTax`

InvBatchDraftToPostedRun

Represents information about the batch job that posts all invoices with the status as `Draft` that are generated by the invoice batch run associated with the billing schedule. This object is available in API version 62.0 and later.

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.

Supported Calls

`describeLayout(),describeSObjects(),getDeleted(),getUpdated(),query(),retrieve(),update()`

Special Access Rules

You need the Billing Operations User permission set to access this object.

Fields

Field	Details
`Comments`	**Type** textarea **Properties** Filter, Nillable, Sort, Update **Description** Additional notes or comments for the invoice batch draft to posted run record.
`CompletionDateTime`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The completion date and time of this invoice batch draft to posted run process.
`CreditApplicationStatus`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** The status of a credit application to the invoices that are posted by the invoice batch draft to posted run process. Valid values are: • `Completed` • `Failed` • `NotApplicable`
`InvBatchDraftToPostedRunNumber`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. The auto-generated number of the invoice batch draft to posted run record.
`InvoiceBatchRunId`	**Type** reference **Properties** Filter, Group, Sort

Field	Details
	Description Required. The ID of the invoice batch run record associated with the invoice batch draft to posted run record. This field is a relationship field. **Relationship Name** InvoiceBatchRun **Relationship Type** Master-detail **Refers To** InvoiceBatchRun (the master object)
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed an invoice batch draft to posted run record indirectly, for example, through a list view or related record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed an invoice batch draft to posted run record. If this value is null, it's possible that the user only accessed the invoice batch draft to posted run record or a related list view (`LastReferencedDate`), but not viewed the invoice batch draft to posted run record itself.
`StartDateTime`	**Type** dateTime **Properties** Filter, Sort **Description** Required. The start date and time of the invoice batch draft to posted run process.
`Status`	**Type** picklist **Properties** Defaulted on create, Filter, Group, Restricted picklist, Sort **Description** Required. The status of the invoice batch draft to posted run record.

Field	Details
	Valid values are: • `Completed` • `CompletedWithErrors` • `Failed` • `Started` The default value is `Started`.
`TotalDraftInvoicesSelected`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The total number of invoices that are selected to be processed by the invoice batch draft to posted run process.
`TotalInvoiceAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total amount of all the invoices posted by the invoice batch draft to posted run process.
`TotalInvoicesFailed`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The total number of invoices that the invoice batch draft to posted run process failed to post.
`TotalPostedInvoices`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The total number of invoices posted by the invoice batch draft to posted run process.

Invoice

Represents information about a financial document describing the total amount a buyer must pay for provided products or services. This object is available in API version 62.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `update()`, `upsert()`

Special Access Rules

You need the Billing Operations User permission set to access this object.

Fields

Field	Details
`Balance`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The outstanding balance of an invoice. The balance is equal to the invoice's total amount with tax, excluding payments and adjustments.
`BillToContactId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The bill to contact ID of the account that's related to an invoice. This field is a relationship field. **Relationship Name** BillToContact **Refers To** Contact
`BillingAccountId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** Required. The ID of the customer account related to an invoice. This field is a relationship field. **Relationship Name** BillingAccount **Refers To** Account

Field	Details
`CorpCrcyCnvTotAmtWithTax`	**Type** double **Properties** Filter, Nillable, Sort, Update **Description** The sum of the total amount with tax on the related invoice lines in corporate currency. Available in API version 63.0 and later.
`CorporateCurrencyCvsnDate`	**Type** date **Properties** Filter, Group, Nillable, Sort, Update **Description** The date on which the invoice's total amount with tax was converted to the corporate currency. Available in API version 63.0 and later.
`CorporateCurrencyCvsnRate`	**Type** double **Properties** Filter, Nillable, Sort, Update **Description** The exchange rate that's used to convert invoice's total amount with tax to the corporate currency. Available in API version 63.0 and later.
`CorporateCurrencyIsoCode`	**Type** string **Properties** Filter, Group, Nillable, Sort, Update **Description** The currency ISO code of the corporate currency. Available in API version 63.0 and later.
`DaysInvoiceOpen`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The number of days from the day an invoice was created to the day it was paid. This field is a calculated field.
`DaysInvoiceOverdue`	**Type** int

Field	Details
	Properties Filter, Group, Nillable, Sort **Description** The number of days since the date when the payment for an invoice was due. This field is a calculated field.
`Description`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Additional details about the invoice.
`DocumentNumber`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. An auto-generated number identifying an invoice.
`DueDate`	**Type** date **Properties** Create, Filter, Group, Sort, Update **Description** Required. The date by when the customer must pay an invoice.
`FullSettlementDate`	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The date when the customer has paid an invoice in full.
`InvBatchDraftToPostedRunId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of the invoice batch to posted run record that generated a posted invoice from a draft invoice.

Field	Details
	This field is a relationship field. **Relationship Name** InvBatchDraftToPostedRun **Refers To** InvBatchDraftToPostedRun
`InvoiceBatchRunId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ID of the invoice batch run record that generated an invoice. This field is a relationship field. **Relationship Name** InvoiceBatchRun **Refers To** InvoiceBatchRun
`InvoiceDate`	**Type** date **Properties** Create, Filter, Group, Sort, Update **Description** Required. The date when an invoice was posted. The payment term's net is added to this date to calculate the invoice's due date. For example, an invoice with an invoice date of 04/01 and the related payment term's period as 30 days has a due date of 05/01.
`InvoiceLockedDateTime`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date and time at which an invoice is locked.
`InvoiceNumber`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Required. The user-specified reference number for the invoice.

Field	Details
`IsBillingScheduleGroupSkipped`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the invoice is generated from a billing transaction directly and if the billing schedule group wasn't created for the transaction (`true`), or if the invoice is generated from the billing schedule group that was created for the transaction (`false`). The default value is `false`. Available in API version 63.0 and later.
`IsInvoiceLocked`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort, Update **Description** Required. Indicates whether an invoice is locked for editing or not (`true`) or not (`false`). The default value is `false`.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed an invoice record indirectly, for example, through a list view or related record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed an invoice record. If this value is null, it's possible that the user only accessed the invoice record or a related list view (`LastReferencedDate`), but not viewed the invoice record itself.
`NetCreditsApplied`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The net credits applied to an invoice. This amount is calculated by subtracting the sum of all the unapplied lines from the sum of all the applied lines.

Field	Details
	This field is a calculated field.
NetPaymentsApplied	**Type** currency **Properties** Filter, Nillable, Sort **Description** The net payments applied to an invoice. This field is visible but isn't used in API version 62.0. This field is a calculated field.
OwnerId	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. The ID of the user who owns an Invoice record. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
PaymentTermId	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The payment term used for an invoice. This field is a relationship field. **Relationship Name** PaymentTerm **Refers To** PaymentTerm
PostedDate	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The date when an invoice was posted.

Field	Details
`ReferenceEntityId`	**Type** reference **Properties** Filter, Group, Nillable, Sort, Update **Description** The ID of the order or order summary that created an invoice. This field is a polymorphic relationship field. **Relationship Name** ReferenceEntity **Refers To** CreditMemo, Order
`SettlementLevel`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** The level at which the credit was applied. Valid values are: • `Invoice` • `InvoiceLine`
`SettlementStatus`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** The status of an invoice's payment. Valid values are: • `Not Applicable` • `Not Settled` • `Partially Settled` • `Settled`
`Status`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. The status of an invoice.

Field	Details
	Valid values are: • `Canceled`— Indicates that the invoice was generated and later canceled. • `Draft`— Indicates that the invoice is a draft. • `Draft In Progress`— Indicates that the draft invoice is in progress. • `Error`— Indicates that an error occurred when processing the invoice. • `Pending`— Indicates that the invoice is being processed. • `Posted`— Indicates that the invoice has been generated and sent to the customer. • `Posting In Progress`—Indicates that the invoice posting is in progress. • `Void In Progress`—Indicates that the invoice is pending a status change. • `Voided`—The invoice's status after the API successfully voids the invoice.
`TotalAdjustmentAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of the invoice's adjustment line amounts. This field is a calculated field.
`TotalAdjustmentAmountWithTax`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of the invoice's adjustment line amounts, including tax.
`TotalAdjustmentTaxAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total amount of tax applied to the invoice line's adjustment lines.
`TotalAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of the total amount on the invoice's lines. This field is a calculated field.

Field	Details
`TotalAmountWithTax`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of the total amount with tax on the invoice’s lines. This field is a calculated field.
`TotalChargeAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of the amount fields on the invoice's charge-type invoice lines. This field is a calculated field.
`TotalChargeAmountWithTax`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of the amount fields on the invoice's charge-type invoice lines, including tax.
`TotalChargeTaxAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total amount of tax applied to the invoice's charge lines.
`TotalConvertedNegAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of all negative invoice lines that were converted to a credit memo. For example, if one negative invoice line was for -$10 and one was for -$15, the total amount that’s converted to a credit memo is -$25. This field is a calculated field.

Field	Details
`TotalTaxAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of the tax amount on the invoice lines. This field is a calculated field.
`UniqueIdentifier`	**Type** string **Properties** Filter, Group, idLookup, Nillable, Sort **Description** The unique identifier of an externally ingested invoice that's used to ensure that duplicate invoices aren't inserted. Available in API version 63.0 and later.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

InvoiceFeed on page 1673
Feed tracking is available for the object.

InvoiceHistory on page 1680
History is available for tracked fields of the object.

InvoiceShare on page 1684
Sharing is available for the object.

InvoiceAddressGroup

Represents the storage of the buyer's address information. This object is available in API version 62.0 and later.

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.

Supported Calls

`delete(), describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), search(), undelete()`

Special Access Rules

You need the Billing Operations User permission set to access this object.

Fields

Field	Details
`Address`	**Type** address **Properties** Filter, Nillable **Description** The buyer's address. See Compound Field Considerations and Limitations for details on geolocation compound fields.
`InvoiceAddressGroupNumber`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. The auto-generated reference number for the invoice address group.
`InvoiceId`	**Type** reference **Properties** Filter, Group, Sort **Description** Required. The ID of the invoice associated with the address group. This field is a relationship field. **Relationship Name** Invoice **Relationship Type** Master-detail **Refers To** Invoice (the master object)

InvoiceBatchRun

Represents a batch processing job in Billing. During an invoice batch run, all billing schedules that meet the specified criteria are processed, resulting in the generation of invoices. This object is available in API version 62.0 and later.

Supported Calls

`describeLayout(),describeSObjects(),getDeleted(),getUpdated(),query(),retrieve(),update()`

Special Access Rules

You need the Billing Admin permission set to access this object.

Fields

Field	Details
`BillingBatchSchedulerId`	**Type** reference **Properties** Filter, Group, Sort **Description** Required. The ID of the related billing batch scheduler. This field is a relationship field. **Relationship Name** BillingBatchScheduler **Refers To** BillingBatchScheduler
`Comments`	**Type** textarea **Properties** Filter, Nillable, Sort, Update **Description** Additional notes or comments for the invoice batch run.
`CompletionTime`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the invoice batch run finished processing.
`CreditApplicationStatus`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** The status of applying credit memos or credit memo lines to invoices or invoice lines during invoices batch runs. Valid values are: • `Completed`

Field	Details
	• `Failed` • `NotApplicable`
`DocGenCompletionTime`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The completion time of generating invoice documents by the invoice batch run. This field is available in API version 63.0 and later with Revenue Cloud Billing.
`DocGenStartTime`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The start time of generating invoice documents by the invoice batch run. This field is available in API version 63.0 and later with Revenue Cloud Billing.
`DocGenStatus`	**Type** picklist **Properties** Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort **Description** The status of the invoice documents generated by the invoice batch run. Valid values are: • `Canceled` • `Completed` • `Failed` • `In Progress` • `Not Started` • `Paused` The default value is `Not Started`. This field is available in API version 63.0 and later with Revenue Cloud Billing.
`InvoiceBatchRunNumber`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort

Field	Details
	Description Required. The auto-generated reference number for the invoice batch run.
`InvoiceDocsGenFailed`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The number of invoice documents that the invoice batch run failed to generate. This field is available in API version 63.0 and later with Revenue Cloud Billing.
`InvoiceDocsGenerated`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The number of invoice documents that the invoice batch run generated. This field is available in API version 63.0 and later with Revenue Cloud Billing.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed an invoice batch run record indirectly, for example, through a list view or related record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed an invoice batch run record. If this value is null, it's possible that the user only accessed the invoice batch run record or a related list view (`LastReferencedDate`), but not viewed the invoice batch run record itself.
`OwnerId`	**Type** reference **Properties** Filter, Group, Sort, Update

Field	Details
	Description Required. The ID of the user who created the related billing batch scheduler. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
`RecoveryStatus`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** The status of the invoice batch run recovery process. Valid values are: • `CompletelyRecovered`—All billing schedules included in the recovery run were reset to `Ready for Invoicing`. These billing schedules are included in the next scheduled invoice batch run. • `PartiallyRecovered`—Some, but not all, billing schedules that were part of the recovery run were reset to `Ready for Invoicing`. The billing schedules that were recovered are included in the next scheduled invoice batch run. The billing schedules that weren't successfully recovered must be manually reset to `Ready for Invoicing` so they can be processed. • `RecoveryFailed`—The recovery job was unsuccessful. • `RecoveryStarted`—The recovery job is in progress.
`StartTime`	**Type** dateTime **Properties** Filter, Sort **Description** Required. The timestamp when the invoice batch run started processing.
`Status`	**Type** picklist **Properties** Defaulted on create, Filter, Group, Restricted picklist, Sort **Description** Required. The status of the invoice batch run. Valid values are: • `Canceled`

Field	Details
	• `Completed` • `Failed` • `Started` • `Stopped` The default value is `Started`.
`StatusSubtype`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** The status subtype of the invoice batch run. The subtype corresponds to the status of the invoice batch run. Valid values are: • `Billing Schedules Filtering Started`—The data processing engine (DPE) selection service has started. The selection service includes the Billing Schedule records that meet the conditions defined when scheduling the invoice run. • `Filter Billing Schedules for Generating Invoices DPE Definition Run In Progress`—The selection DPE was created and initiated for execution. • `Filter Billing Schedules for Generating Invoices DPE Definition Run Completed`—The selection DPE was completed and a message was sent to the platform events handler. • `Invoice Generation Started`—The invoice generation service has started. • `Generate Invoices for Billing Schedules DPE Definition Run In Progress`—The invoice generation DPE was created and initiated for execution. • `Generate Invoices for Billing Schedules DPE Definition Run Completed`—The invoice generation DPE was completed and a message was sent to the platform events handler. • `Invoice Tax Calculation In Progress`—The tax calculation step has started. • `Billing Schedule and Billing Schedule Group Updates In Progress`—The post processing step has started. • `Invoice Generation Summarization In Progress`—The invoice generation summarization step has started. • `Completed`—The invoice batch run has completed successfully. • `Failed`—The invoice batch run was unsuccessful. Available in API version 63.0 and later.

Field	Details
`TotBillSchdUpdtDurDrftToPost`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The total duration taken to update the billing schedule from `Draft` to `Posted` status during an invoice batch run.
`TotalBillSchedRecovered`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The number of billing schedules that were a part of the recovery run that were reset to `Ready for Invoicing`. These billing schedules are included in the next scheduled invoice batch run.
`TotalBillSchedUnrecovered`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The number of billing schedules that were a part of the recovery run that weren't reset to `Ready for Invoicing`. These billing schedules that weren't successfully recovered must be manually reset to `Ready for Invoicing` so they can be processed.
`TotalBillingSchedulesFailed`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The total number of billing schedules that weren't successfully processed. When a billing schedule isn't successfully processed, then the system doesn't generate an invoice for it. For details about errors, check the Revenue Transaction Error Log.
`TotalBsSuccessfullyProcessed`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The total number of billing schedules for which the system was able to generate and process invoices.

Field	Details
`TotalDraftInvoiceAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total amount of all invoices in the `Draft` status.
`TotalDraftInvoices`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The total number of invoices in the `Draft` status.
`TotalFilteredBillingSchedules`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The total number of billing schedules that met the invoice run scheduler's matching criteria. The matching criteria specify which billing schedules are included in the invoice batch run. The field label is **Total Matching Billing Schedules**.
`TotalInvSuccessfullyProcessed`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The total number of invoices that were successfully processed.
`TotalInvoicedAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total amount of income including taxes that's represented by the successfully processed invoices.
`TotalInvoicesCanceled`	**Type** int **Properties** Filter, Group, Nillable, Sort

Field	Details
	Description The total number of invoices that weren't processed. To find out what went wrong, check the Revenue Transaction Error Log. Then fix the errors and run the invoice batch run recovery process.
`TotalInvoicesFailed`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The total number of invoices that weren't processed successfully. To find out what went wrong, check the Revenue Transaction Error Log. Then fix the errors and run the invoice batch run recovery process.
`TotalInvoicesGenerated`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The total number of invoices that were generated from the billing schedules processed by the invoice batch run.
`TotalPostedInvoices`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The total number of invoices in `Posted` status that were generated from the billing schedules.

InvoiceBatchRunCriteria

Represents a batch processing job and its required criteria in Billing. During an invoice batch run, all billing schedules that meet the specified criteria are processed, resulting in the generation of invoices. This object is available in API version 62.0 and later.

Supported Calls

`describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

You need the Billing Admin permission set to access this object.

Fields

Field	Details
`Comments`	**Type** textarea **Properties** Filter, Nillable, Sort **Description** Additional notes or comments for the invoice batch run criteria.
`CriteriaExpression`	**Type** textarea **Properties** Filter, Nillable, Sort **Description** The formula that specifies criteria for filtering the billing schedules. For example, you can filter billing schedules by the currency code.
`CriteriaMatchType`	**Type** picklist **Properties** Defaulted on create, Filter, Group, Restricted picklist, Sort **Description** Required. The type of matching criteria required for the batch. Valid value is `MatchAll`. The default value is `MatchAll`.
`ExpectedInvoiceStatus`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** The type of invoice a batch run generates. Valid values are: • `Draft` • `Posted`
`InvoiceBatchRunCriteriaNumber`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort

Field	Details
	Description Required. The auto-generated reference number for the invoice batch run criteria.
`InvoiceDate`	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The date displayed on the invoice. This date is also used for tax calculations.
`InvoiceDateOffset`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The offset that's applied to the target date to calculate the invoice date.
`IsInvoiceDateFromRunDate`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Required. Indicates whether the invoice date is derived from the run date (`true`) or not (`false`). The default value is `false`. Available in API version 63.0 and later.
`OwnerId`	**Type** reference **Properties** Filter, Group, Sort **Description** Required. The ID of the user who created the invoice batch run criteria. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
`TargetDate`	**Type** date

Field	Details
	Properties Filter, Group, Nillable, Sort **Description** The target date for the invoice run. Billing schedules having the next billing date before this date are picked up for invoicing.
`TargetDateOffset`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The offset that's applied to the next run date to calculate the target date.

InvoiceBatchRunRecovery

Represents information about the recovery procedure of an invoice batch run. This object is available in API version 62.0 and later.

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.

Supported Calls

`describeLayout(),describeSObjects(),getDeleted(),getUpdated(),query(),retrieve(),update()`

Special Access Rules

You need the Billing Admin permission set to access this object.

Fields

Field	Details
`Comments`	**Type** textarea **Properties** Filter, Nillable, Sort, Update **Description** Additional notes or comments for the invoice batch run recovery.
`CompletionTime`	**Type** dateTime

Field	Details
	Properties Filter, Nillable, Sort **Description** The timestamp when the invoice batch run recovery finished processing.
`InvoiceBatchRunId`	**Type** reference **Properties** Filter, Group, Sort **Description** Required. A unique identifier of the invoice batch run that's related to this recovery run. This field is a relationship field. **Relationship Name** InvoiceBatchRun **Relationship Type** Master-detail **Refers To** InvoiceBatchRun (the master object)
`InvoiceBatchRunRecoveryNumber`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. A unique identifier of the recovery process for the invoice batch run.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed an invoice batch run recovery record indirectly, for example, through a list view or related record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed an invoice batch run recovery record. If this value is null, it’s possible that the user only accessed the invoice batch run recovery

Field	Details
	record or a related list view (`LastReferencedDate`), but not viewed the invoice batch run recovery record itself.
`StartTime`	**Type** dateTime **Properties** Filter, Sort **Description** Required. The timestamp when the invoice batch run recovery started processing.
`Status`	**Type** picklist **Properties** Defaulted on create, Filter, Group, Restricted picklist, Sort **Description** Required. The final state of the recovery process for the invoice batch runs. Valid values are: • `Completed` • `CompletedWithErrors` • `Failed` • `Started` The default value is `Started`.

InvoiceDocument

Represents the PDF document generated for an invoice. This object is available in API version 63.0 and later.

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.

Supported Calls

`describeLayout(),describeSObjects(),getDeleted(),getUpdated(),query(),retrieve(),search()`

Special Access Rules

You need Revenue Cloud Billing, and the Billing Admin permission set or the Billing Operations User permission set to access this object.

Fields

Field	Details
DateGenerated	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The date on which the PDF is generated.
DocumentNumber	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. A unique number assigned to the PDF invoice document.
ErrorMessage	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** Any errors that occur during PDF generation.
InvoiceId	**Type** reference **Properties** Filter, Group, Sort **Description** Required. The ID of the Invoice to which the Invoice Doc is attached. This field is a relationship field. **Relationship Name** Invoice **Relationship Type** Master-detail **Refers To** Invoice (the master object)
Status	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort

Field	Details
	Description The status of the PDF generation process. Valid values are: • `Blocked` • `Cancelled` • `Failure` • `Pending` • `Success`

InvoiceLine

Represents the amount that a buyer must pay for a product, service, or fee. Invoice lines are created based on the amount of an order line. This object is available in API version 62.0 and later.

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.

Supported Calls

`describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `update()`

Special Access Rules

You need the Billing Admin permission set to access this object.

Fields

Field	Details
`AdjustmentAmount`	**Type** currency **Properties** Filter, Nillable, Sort, Update **Description** The sum of adjustments made to the invoice line.
`AdjustmentAmountWithTax`	**Type** currency **Properties** Filter, Nillable, Sort

Field	Details
	Description The sum of adjustment amounts, including associated taxes related to the invoice line.
`AdjustmentTaxAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The sum of tax adjustments to the invoice line.
`Balance`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The outstanding balance for an invoice line. This amount is equal to the invoice's total amount with tax after deducting the payments made.
`BillingAddressId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** This field is related to an `InvoiceAddressGroup` field containing the billing address for the invoice line. For example, one `InvoiceAddressGroup` field is related to the invoiceLine's `BillingAddressID` field, and another `InvoiceAddressGroup` field is related to the invoiceLine's `ShippingAddressId` field. This field is a relationship field. **Relationship Name** BillingAddress **Refers To** InvoiceAddressGroup
`BillingScheduleGroupId`	**Type** reference **Properties** Filter, Group, Nillable, Sort, Update **Description** The ID of the billing schedule group for the invoice line. This field is a relationship field.

Field	Details
	Relationship Name BillingScheduleGroup **Refers To** BillingScheduleGroup
`BillingScheduleId`	**Type** reference **Properties** Filter, Group, Nillable, Sort, Update **Description** The ID of the billing schedule for the invoice line. This field is a relationship field. **Relationship Name** BillingSchedule **Refers To** BillingSchedule
`ChargeAmount`	**Type** currency **Properties** Filter, Nillable, Sort, Update **Description** The sum of charges made to the invoice line.
`ChargeAmountWithTax`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount on a charge invoice line, including tax.
`ChargeConvertedNegAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount on a charge invoice line that's converted to credit.
`ChargeTaxAmount`	**Type** currency

Field	Details
	Properties Filter, Nillable, Sort **Description** The tax to be applied on a charge invoice line.
`ConvertedNegAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount from an invoice line that's converted to credit.
`CorpCurrencyCnvChargeAmt`	**Type** double **Properties** Filter, Nillable, Sort, Update **Description** The sum of charges made to the invoice line in corporate currency. Available in API version 63.0 and later.
`CorpCurrencyCnvTotalTaxAmt`	**Type** double **Properties** Filter, Nillable, Sort, Update **Description** The total tax amount of the invoice line in corporate currency. Available in API version 63.0 and later.
`CorporateCurrencyCvsnDate`	**Type** date **Properties** Filter, Group, Nillable, Sort, Update **Description** The date on which the invoice line amounts are converted to corporate currency. Available in API version 63.0 and later.
`CorporateCurrencyCvsnRate`	**Type** double **Properties** Filter, Nillable, Sort, Update

Field	Details
	Description The exchange rate that's used to convert the invoice line amounts to corporate currency. Available in API version 63.0 and later.
`CorporateCurrencyIsoCode`	**Type** string **Properties** Filter, Group, Nillable, Sort, Update **Description** The currency ISO code of the corporate currency. Available in API version 63.0 and later.
`Description`	**Type** string **Properties** Filter, Group, Nillable, Sort, Update **Description** The description of the invoice line.
`HasMultipleItems`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Required. Indicates whether this field merges items from the same billing period (`true`) or not (`false`). The default value is `false`.
`InvoiceId`	**Type** reference **Properties** Filter, Group, Sort **Description** Required. The ID of the invoice record that contains this invoice line. This field is a relationship field. **Relationship Name** Invoice **Relationship Type** Master-detail **Refers To** Invoice (the master object)

Field	Details
`InvoiceLineEndDate`	**Type** date **Properties** Filter, Group, Sort, Update **Description** Required. The end date of the billing for the service for invoice lines made from a time-based service.
`InvoiceLineStartDate`	**Type** date **Properties** Filter, Group, Sort, Update **Description** Required. The first date of the billing for the service for invoice lines made from a time-based service.
`InvoiceStatus`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The status of the invoice line. This field is inherited from the invoice's status.
`IsUsageBasedInvoiceLine`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Required. Indicates whether the product is usage-based (`true`) or not (`false`). The default value is `false`. This field is available in API version 63.0 and later with Revenue Cloud Billing.
`LegalEntityAccountingPeriodId`	**Type** reference **Properties** Filter, Group, Nillable, Sort, Update **Description** The ID of the legal entity accounting period record used in this invoice line. This field is a relationship field. **Relationship Name** LegalEntityAccountingPeriod

Field	Details
	Refers To LegalEntyAccountingPeriod
`LegalEntityId`	**Type** reference **Properties** Filter, Group, Nillable, Sort, Update **Description** The ID of the legal entity record used in this invoice line. This field is a relationship field. **Relationship Name** LegalEntity **Refers To** LegalEntity
`LineAmount`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The amount of the invoice line. This field is a calculated field.
`Name`	**Type** string **Properties** Filter, Group, idLookup, Sort, Update **Description** Required. The name of the invoice line.
`NetCreditsApplied`	**Type** currency **Properties** Filter, Nillable, Sort **Description** The total credit memo line amount applied to the invoice line. This amount is calculated by subtracting the unapplied credit memo line amount from the applied credit memo line amount. This field is a calculated field.

Field	Details
OverageUnitOfMeasure	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The unit that's used to measure the overage. For example, byte or minute. This field is available in API version 63.0 and later with Revenue Cloud Billing.
ParentInvoiceLineId	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of the parent invoice line record. This field is a relationship field. **Relationship Name** ParentInvoiceLine **Refers To** InvoiceLine
Product2Id	**Type** reference **Properties** Filter, Group, Nillable, Sort, Update **Description** The ID of the product that was charged or ordered to create the invoice line. This field is a relationship field. **Relationship Name** Product2 **Refers To** Product2
Quantity	**Type** double **Properties** Filter, Nillable, Sort, Update **Description** The number of units of the order product that created the invoice line.
ReferenceEntityItemId	**Type** reference

Field	Details
	Properties Filter, Group, Nillable, Sort, Update **Description** The order item or adjustment item that created the invoice line. This field is a polymorphic relationship field. **Relationship Name** ReferenceEntityItem **Refers To** OrderItem, OrderItemAdjustmentLineItem
`ReferenceEntityItemType`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The type of transaction that created the invoice line. Valid values are: • `Delivery Charge` • `Fee` • `Order Product`
`ReferenceEntityItemTypeCode`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The type of object that created the invoice line. Valid values are: • `Charge` • `Product`
`ShippingAddressId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of the shipping address record associated with the invoice line. This field is a relationship field. **Relationship Name** ShippingAddress

Field	Details
	Refers To InvoiceAddressGroup
`TaxAmount`	**Type** currency **Properties** Filter, Nillable, Sort, Update **Description** The total tax for the invoice line.
`TaxTreatmentId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The tax treatment used on this invoice line. This field is a relationship field. **Relationship Name** TaxTreatment **Refers To** TaxTreatment
`UnitOfMeasureId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The unit of measure of the product associated with the invoice line. Available in API version 63.0 and later. This field is a relationship field. **Relationship Name** UnitOfMeasure **Refers To** UnitOfMeasure
`UnitPrice`	**Type** currency **Properties** Filter, Nillable, Sort, Update **Description** The price for one unit of the item on the invoice line.

Field	Details
`UsageOverageQuantity`	**Type** double **Properties** Filter, Nillable, Sort **Description** The quantity of the usage overage that's being invoiced. This field is available in API version 63.0 and later with Revenue Cloud Billing.
`UsageProductBillSchdGrpId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of the usage-based product billing schedule group associated with the invoice line. This field is available in API version 63.0 and later with Revenue Cloud Billing. This field is a relationship field. **Relationship Name** UsageProductBillSchdGrp **Refers To** BillingScheduleGroup
`UsageProductId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of the usage-based product that was charged or ordered to create the invoice line. This field is available in API version 63.0 and later with Revenue Cloud Billing. This field is a relationship field. **Relationship Name** UsageProduct **Refers To** Product2

InvoiceLineRelationship

Represents a relationship between invoice line items to support bundles where one parent invoice line has multiple child invoice lines. This object is available in API version 62.0 and later.

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.

Supported Calls

`describeLayout(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

You need the Billing Admin permission set to access this object.

Fields

Field	Details
`AssociatedInvoiceLineId`	**Type** reference **Properties** Filter, Group, Sort **Description** Required. The ID of the related invoice line record. In a bundle relationship, this invoice line is the child. This field is a relationship field. **Relationship Name** AssociatedInvoiceLine **Refers To** InvoiceLine
`AssociatedInvoiceLinePricing`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Required. This field describes how the related invoice line is priced relative to the primary invoice line. Valid values are: • `IncludedInBundlePrice` • `NotIncludedInBundlePrice`
`AssociatedInvoiceLineRole`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Required. This field describes the role of the related invoice line in the relationship. The value is derived from the `AssociatedProductRoleCat` field of the ProductRelationshipType object.

Field	Details
	Valid values are: • `AddOnComponent` • `BundleComponent` • `ClassificationComponent` • `SetComponent`
`InvoiceId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of the related invoice record. This field is a relationship field. **Relationship Name** Invoice **Refers To** Invoice
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed an invoice line relationship record indirectly, for example, through a list view or related record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed an invoice line relationship record. If this value is null, it's possible that the user only accessed the invoice line relationship record or a related list view (`LastReferencedDate`), but not viewed the invoice line relationship record itself.
`MainInvoiceLineId`	**Type** reference **Properties** Filter, Group, Sort

Field	Details
	Description Required. The ID of the primary invoice line record. In a bundle relationship, this invoice line is the parent. This field is a relationship field. **Relationship Name** MainInvoiceLine **Relationship Type** Master-detail **Refers To** InvoiceLine (the master object)
`MainInvoiceLineRole`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Required. This field describes the role of the primary invoice line in the relationship. The value is derived from the `MainProductRoleCat` field of the ProductRelationshipType object. Valid values are: • `AddOn` • `Bundle` • `Set`
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. An auto-generated number identifying the invoice line relationship.
`ProductRelationshipTypeId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of the product relationship type record between the main and associated invoice lines. This field is a relationship field. **Relationship Name** ProductRelationshipType

Field	Details
	Refers To ProductRelationshipType

InvoiceLineTax

Represents tax information of an invoice line of type `Tax`. This object is available in API version 62.0 and later.

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.

Supported Calls

`describeLayout(),describeSObjects(),getDeleted(),getUpdated(),query(),retrieve(),search(),update()`

Special Access Rules

You need the Tax Admin permission set to access this object.

Fields

Field	Details
`BillingAddressId`	**Type** reference **Properties** Filter, Group, Nillable, Sort, Update **Description** The billing address of the parent invoice line. This field is a relationship field. **Relationship Name** BillingAddress **Refers To** InvoiceAddressGroup
`ConvertedNegAmount`	**Type** currency **Properties** Filter, Nillable, Sort, Update **Description** The amount from the parent invoice line that's converted to credit.

Field	Details
`CorpCrcyCnvTaxAmount`	**Type** double **Properties** Filter, Group, Nillable, Sort **Description** The total tax amount of the invoice line tax in corporate currency. Available in API version 63.0 and later.
`CorporateCurrencyCvsnDate`	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** The date on which the invoice line tax amounts are converted to the corporate currency. Available in API version 63.0 and later.
`CorporateCurrencyCvsnRate`	**Type** double **Properties** Filter, Nillable, Sort **Description** The exchange rate that's used to convert the invoice line tax amounts to the corporate currency. Available in API version 63.0 and later.
`CorporateCurrencyIsoCode`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The currency ISO code of the corporate currency. Available in API version 63.0 and later.
`Description`	**Type** string **Properties** Filter, Group, Nillable, Sort, Update **Description** Additional details about the invoice line tax.
`EndDate`	**Type** date **Properties** Filter, Group, Sort, Update

Field	Details
	Description Required. The end date of an invoice line tax.
`InvoiceLineId`	**Type** reference **Properties** Filter, Group, Sort **Description** Required. The ID of the parent charge or adjustment invoice line record that an invoice line tax is related to. This field is a relationship field. **Relationship Name** InvoiceLine **Relationship Type** Master-detail **Refers To** InvoiceLine (the master object)
`InvoiceLineTaxNumber`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. An auto-generated number identifying an invoice line tax.
`LegalEntityAccountingPeriodId`	**Type** reference **Properties** Filter, Group, Nillable, Sort, Update **Description** The ID of the legal entity accounting period record that's related to an invoice line tax. This field is a relationship field. **Relationship Name** LegalEntityAccountingPeriod **Refers To** LegalEntyAccountingPeriod
`LegalEntityId`	**Type** reference **Properties** Filter, Group, Nillable, Sort, Update

Field	Details
	Description The ID of the legal entity that's related to an invoice line tax. This field is a relationship field. **Relationship Name** LegalEntity **Refers To** LegalEntity
`Product2Id`	**Type** reference **Properties** Filter, Group, Nillable, Sort, Update **Description** The product that was charged or ordered to create the parent invoice line. Available in API version 62.0 only. This field is a relationship field. **Relationship Name** Product2 **Refers To** Product2
`Quantity`	**Type** double **Properties** Filter, Nillable, Sort, Update **Description** The number of units of the order product that created the parent invoice line. Available in API version 62.0 only.
`ShippingAddressId`	**Type** reference **Properties** Filter, Group, Nillable, Sort, Update **Description** The ID of the shipping address record of the parent invoice line. This field is a relationship field. **Relationship Name** ShippingAddress **Refers To** InvoiceAddressGroup

Field	Details
`StartDate`	**Type** date **Properties** Filter, Group, Sort, Update **Description** Required. The start date of an invoice line tax.
`TaxAmount`	**Type** currency **Properties** Filter, Nillable, Sort, Update **Description** The total amount of an invoice line tax. The tax exempt amount is included in the tax amount.
`TaxCode`	**Type** string **Properties** Filter, Group, Nillable, Sort, Update **Description** The code that's used to calculate the tax rate for the parent invoice line.
`TaxDocumentNumber`	**Type** string **Properties** Filter, Group, Nillable, Sort, Update **Description** The number of the latest record in the external tax engine in which the parent invoice line is included.
`TaxEffectiveDate`	**Type** date **Properties** Filter, Group, Nillable, Sort, Update **Description** The date used to calculate the tax amount.
`TaxExemptAmount`	**Type** currency **Properties** Filter, Nillable, Sort, Update

Field	Details
	Description The amount that's exempted from tax.
`TaxName`	**Type** string **Properties** Filter, Group, Nillable, Sort, Update **Description** The user-defined name for an invoice line tax.
`TaxProcessingStatus`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** The processing status of the invoice line tax. Valid values are: • `Estimated` • `Pending` • `Posted`
`TaxRate`	**Type** percent **Properties** Filter, Nillable, Sort, Update **Description** The percentage of the order product price that's used to calculate tax.
`TaxTransactionNumber`	**Type** string **Properties** Filter, Group, Nillable, Sort, Update **Description** The number of the transaction in the external tax engine that calculated tax for the parent invoice line.
`TaxTreatmentId`	**Type** reference **Properties** Filter, Group, Nillable, Sort, Update

Field	Details
	Description The ID of the tax treatment record that's related to an invoice line tax. This field is a relationship field. **Relationship Name** TaxTreatment **Refers To** TaxTreatment
`UnitPrice`	**Type** currency **Properties** Filter, Nillable, Sort, Update **Description** The price per unit of the order product that's related to an invoice line tax. Available in API version 62.0 only.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

InvoiceLineTaxFeed on page 1673
Feed tracking is available for the object.

InvoiceLineTaxHistory on page 1680
History is available for tracked fields of the object.

LegalEntity

Represents the way an organization is structured. An organization can be a single legal entity or it can comprise more than one legal entity. This object is available in API version 62.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

You need the Accounts Receivables Admin permission set to access this object.

Fields

Field	Details
`CompanyName`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The name of the company that this legal entity represents.
`Description`	**Type** textarea **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Additional details about the legal entity.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a legal entity indirectly, for example, through a list view or related record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a legal entity. If this value is null, it's possible that the user only accessed the legal entity or a related list view (`LastReferencedDate`), but not viewed the legal entity itself.
`LegalEntityAddress`	**Type** address **Properties** Filter, Nillable **Description** The address of the company that this legal entity represents. See Address Compound Fields for details on compound address fields.

Field	Details
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** Required. The name of the legal entity.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. The ID of the record owner. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
`Status`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The status of the legal entity. Valid values are: • `Active` • `Inactive`

LegalEntyAccountingPeriod

Represents a junction between a legal entity and an accounting period. This object is available in API version 62.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

You need the Accounts Receivables Admin permission set to access this object.

Fields

Field	Details
`AccountingPeriodId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ID of the accounting period record that's related to a legal entity accounting period. This field is a relationship field. **Relationship Name** AccountingPeriod **Refers To** AccountingPeriod
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a legal entity accounting period record indirectly, for example, through a list view or related record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a legal entity accounting period record. If this value is null, it's possible that the user only accessed the legal entity accounting period record or a related list view (`LastReferencedDate`), but not viewed the legal entity accounting period record itself.
`LegalEntityId`	**Type** reference **Properties** Create, Filter, Group, Sort, Update **Description** Required. The ID of the legal entity record that's related to a legal entity accounting period. This field is a relationship field. **Relationship Name** LegalEntity

Field	Details
	Refers To LegalEntity
`Name`	**Type** string **Properties** Filter, Group, idLookup, Sort **Description** Required. The auto-generated name of a legal entity accounting period. The name is a combination of the names of the legal entity and the accounting period.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. The ID of the user who owns a legal entity accounting period record. This field is a polymorphic relationship field. **Relationship Name** Owner **Refers To** Group, User
`Status`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort **Description** Required. The status of a legal entity accounting period record. Valid values are: • `Closed` • `Error` • `Open` • `PendingClosure` • `PendingReopen` • `Reopened`

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

LegalEntyAccountingPeriodShare on page 1684
Sharing is available for the object.

PaymentTerm

Represents an agreement between a buyer and a seller about when payment is due for an invoice. This object is available in API version 62.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

You need the Billing Admin permission set to access this object.

Fields

Field	Details
`Description`	**Type** textarea **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Additional details about the payment term.
`IsDefault`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. Indicates whether the payment term is the default term for your Salesforce org (`true`) or not (`false`). The default value is `false`.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort

Field	Details
	Description The timestamp when the current user last accessed this record indirectly, for example, through a list view or related record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed this record or list view. If this value is null, and `LastReferenceDate` field is not null, the user accessed this record or list view indirectly.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** Required. The name of the payment term. This name appears on the invoice.
`Status`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. Specifies whether the payment term is available for use on invoices. Possible values are: • `Active` • `Draft` • `Inactive` The default value is `Draft`.

PaymentTermItem

Represents configuration of a payment term. This object is available in API version 62.0 and later.

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.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

You need the Billing Admin permission set to access this object.

Fields

Field	Details
`Description`	**Type** textarea **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Additional details about the payment term item.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed this record indirectly, for example, through a list view or related record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed this record or list view. If this value is null, and `LastReferenceDate` field is not null, the user accessed this record or list view indirectly.
`Name`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. The autogenerated name of the payment term item.

Field	Details
`PaymentTermId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** Required. The ID of the parent payment term record. This field is a relationship field. **Relationship Name** PaymentTerm **Relationship Type** Master-detail **Refers To** PaymentTerm (the master object)
`PaymentTimeframe`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. The period by when the payment is expected. Valid value is: • `Standard`— The payment must be made by the date specified in the payment term. If payment isn't received by the due date, the payment becomes overdue.
`Period`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The number of units in the payment period. Used with the `PeriodUnit` field. For example, to define a payment term of Net 30, enter 30 as the value of the `Period` field and select `Days` as the value of the `PeriodUnit` field.
`PeriodUnit`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The unit of time for the payment period. For example, to define a payment term of Net 30, enter 30 as the value of the `Period` field and select `Days` as the value of the `PeriodUnit` field.

Field	Details
	The valid value is: • `Days`
`Type`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. Specifies how the payment term and invoice due date are derived. Valid values are: • `EOM-Plus-X`—Calculates the invoice due date by adding the `Period` field value to the last day of the month in which the invoice is posted. Available in API version 63.0 and later. • `Period-Based`—Calculates the invoice due date by adding the `Period` field value to the invoice date.

RevenueTransactionErrorLog

Represents the details of errors that occurred during the processing of a request. The error record persists until a new error with the same category, primary record, and, if necessary, related record occurs. This object is available in API version 62.0 and later.

Supported Calls

`describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

Special Access Rules

You need the Billing Operations User permission set to access this object.

Fields

Field	Details
`AsyncOperationTrackerId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of the async operation tracker record created by the request. Async operation tracker records contain information about the status of the asynchronous process initiated by the request. This field is a relationship field.

Field	Details
	Relationship Name AsyncOperationTracker **Refers To** AsyncOperationTracker
`Category`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** Specifies the context of the source of error.
`ErrorCode`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The error code shown during the request processing, such as INVALID_INPUT.
`ErrorLogNumber`	**Type** string **Properties** Autonumber, Defaulted on create, Filter, idLookup, Sort **Description** Required. An auto-generated unique ID that identifies the error.
`ErrorMessage`	**Type** textarea **Properties** Create **Description** This field contains information about the error and how to resolve it.
`OwnerId`	**Type** reference **Properties** Filter, Group, Sort **Description** Required. The ID of the user who made the request that resulted in the creation of the error log. This field is a polymorphic relationship field.

Field	Details
	Relationship Name Owner **Refers To** Group, User
PrimaryRecord2Id	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of the record that's associated with this error. This field is a polymorphic relationship field. **Relationship Name** PrimaryRecord2 **Refers To** LegalEntyAccountingPeriod
PrimaryRecordId	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of the record that's associated with this error. For example, if the error occurred while creating an invoice from an order, the primary ID is the ID of the order. This field is a polymorphic relationship field. **Relationship Name** PrimaryRecord **Refers To** Asset, AsyncOperationTracker, BillingBatchScheduler, BillingSchedule, CreditMemo, InvBatchDraftToPostedRun, Invoice, InvoiceBatchRun, InvoiceBatchRunRecovery, Order, OrderItem, RevenueAsyncOperation, RuleLibraryVersion
PrimaryTextRecord	**Type** string **Properties** Create, Filter, Group, Nillable, Sort **Description** The identifier of the primary record associated with the error log. Available in API version 63.0 and later.

Field	Details
`RelatedRecord2Id`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of a record that can provide additional context about the error. For example, if `PrimaryRecordId` is the ID of an order, this field could be the ID of an invoice line. This field is a polymorphic relationship field. **Relationship Name** RelatedRecord2 **Refers To** CreditMemoLine, CreditMemoLineInvoiceLine, CreditMemoLineTax, InvoiceLine, InvoiceLineTax
`RelatedRecordId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of a record that can provide additional context about the error. For example, if `PrimaryRecordId` is the ID of an order, this field could be the ID of an order item. This field is a polymorphic relationship field. **Relationship Name** RelatedRecord **Refers To** Asset, BillingBatchScheduler, BillingSchedule, BillingScheduleGroup, CreditMemo, CreditMemoLine, InvBatchDraftToPostedRun, Invoice, InvoiceBatchRun, InvoiceLine, OrderItem
`RequestIdentifier`	**Type** string **Properties** Filter, Group, idLookup, Nillable, Sort **Description** A unique ID returned by the request. Use this ID to identify the revenue transaction error log records for a specific request.

TaxEngine

Represents information about an instance of a tax engine provider as well as the merchant credentials for that specific instance. This object is available in API version 62.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

You need the Tax Admin permission set to access this object.

Fields

Field	Details
`Description`	**Type** textarea **Properties** Create, Filter, Nillable, Sort, Update **Description** The description of an instance of the tax engine provider and merchant credential.
`ExternalReference`	**Type** string **Properties** Create, Filter, Group, idLookup, Nillable, Sort, Update **Description** Information about the external platform used for the tax engine.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a tax engine record indirectly, for example, through a list view or related record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a tax engine record. If this value is null, it's possible that the user only accessed the tax engine record or a related list view (`LastReferencedDate`), but not viewed the tax engine record itself.

Field	Details
`MerchantCredentialId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Required. The merchant credential setup object in Salesforce. The Tax Calculation API sends the merchant credential's information to the external tax engine that's used for tax calculation. This field is a relationship field. **Relationship Name** MerchantCredential **Refers To** NamedCredential
`SellerCode`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Required. The seller code of the transaction for which the tax engine integration log was captured.
`Status`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. The status of the tax engine record. Valid values are: • `Active` • `Inactive`
`TaxEngineAddress`	**Type** address **Properties** Filter **Description** Required. The compound form of the tax engine address. This field is used when the request doesn't contain a Ship To address. See Address Compound Fields for details on compound address fields. This field is a read-only field.

Field	Details
`TaxEngineName`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** Required. The name of the tax engine record.
`TaxEngineProviderId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Required. The tax engine provider that's related to the tax engine record. This field is a relationship field. **Relationship Name** TaxEngineProvider **Refers To** TaxEngineProvider
`TaxPrvdAccountIdentifier`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The unique identifier of the external tax provider's account. Available in API version 63.0 and later.
`Type`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The type of the tax engine that's used to calculate tax. Valid values are: • `CommerceTaxExtension` • `RevenueCloudTaxExtension` • `StandardTaxEngine` • `StripeNative` Available in API version 63.0 and later.

TaxEngineProvider

Represents general information about a service that manages a tax engine. Tax engine providers have a one-to-many relationship with tax engines, where the tax engine record represents a specific configuration of a tax engine that can be assigned to multiple order items. This object is available in API version 62.0 and later.

Supported Calls

`create(),delete(),describeLayout(),describeSObjects(),query(),retrieve(),update(),upsert()`

Special Access Rules

You need the Tax Admin permission set to access this object.

Fields

Field	Details
`ApexAdapterId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ID of the Apex adapter used by the tax provider. This field is unique within your organization. This field is a relationship field. **Relationship Name** ApexAdapter **Refers To** ApexClass
`Description`	**Type** textarea **Properties** Create, Filter, Nillable, Sort, Update **Description** Additional details about the tax engine provider.
`DeveloperName`	**Type** string **Properties** Create, Filter, Group, Sort, Update **Description** Required. The API name for the tax engine provider record.

Field	Details
`Language`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The language used by the tax engine provider. Values appear based on their language codes in Salesforce, such as `da` for Danish or `th` for Thai.
`MasterLabel`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** Required. The label used for the tax engine's API in Salesforce.
`NamespacePrefix`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The Apex namespace prefix of the API used for the tax engine.

TaxPolicy

Represents information about a group of tax treatments, where each treatment represents parameters to determine how a particular product is taxed for a transaction line item. Tax policies are related to products, which pass the policy on to the resulting order items and in turn the billing schedules. This object is available in API version 62.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

You need the Tax Admin permission set to access this object.

Fields

Field	Details
`DefaultTaxTreatmentId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ID of the default tax treatment record. When you order a product, the order product, and in turn the billing schedule receives this tax treatment. This field is a relationship field. **Relationship Name** DefaultTaxTreatment **Refers To** TaxTreatment
`Description`	**Type** textarea **Properties** Create, Filter, Nillable, Sort, Update **Description** Additional details about the tax policy.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed an invoice batch draft to posted run record indirectly, for example, through a list view or related record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last viewed a tax policy record. If this value is null, it's possible that the user only accessed the tax policy record or a related list view (`LastReferencedDate`), but not viewed the tax policy record itself.
`Name`	**Type** string

Field	Details
	Properties Create, Filter, Group, idLookup, Sort, Update **Description** Required. The name of the tax policy.
`Status`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. The status of the tax policy. To calculate tax for order products, products must have an active tax policy. Tax policies are created with a `Draft` status before being assigned to a product or order product. After activating a tax policy, you can't edit certain policy fields. Valid values are: • `Active` • `Draft` • `Inactive`
`TreatmentSelection`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. Specifies the selection of tax treatments to billing schedules that are related to the tax policy. Valid values are: • Default —Billing schedules receive the tax treatment defined in the tax policy's `DefaultTreatmentId` field. • `LegalEntity`—Billing schedules receive the tax treatment based on matching legal entities between itself and the tax treatment. • `Manual`—Billing schedules don't receive tax treatments based on the tax policy. You must specify the treatment.

TaxTreatment

Represents information about tax calculation by external engines. Each product requires a tax policy to determine whether to apply tax. Each tax policy requires at least one tax treatment. The tax treatments determine how taxable products are taxed. This object is available in API version 62.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

You need the Tax Admin permission set to access this object.

Fields

Field	Details
`Description`	**Type** textarea **Properties** Create, Filter, Nillable, Sort, Update **Description** Additional details about the tax treatment.
`IsTaxable`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. Indicates whether to calculate tax for the order items covered by the tax treatment (`true`) or not (`false`). When this value is `true`, the CalculateTax API is called for the order item during the order item's creation. The default value is `false`.
`LastReferencedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The timestamp when the current user last accessed a tax treatment record indirectly, for example, through a list view or related record.
`LastViewedDate`	**Type** dateTime **Properties** Filter, Nillable, Sort

Field	Details
	Description The timestamp when the current user last viewed a tax treatment record. If this value is null, it's possible that the user only accessed the tax treatment record or a related list view (`LastReferencedDate`), but not viewed the tax treatment record itself.
`LegalEntityId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ID of the legal entity record that's related to the tax treatment. This field is a relationship field. **Relationship Name** LegalEntity **Refers To** LegalEntity
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** Required. The name of the tax treatment.
`ProductCode`	**Type** string **Properties** Create, Filter, Group, idLookup, Nillable, Sort, Update **Description** The code of the product that the tax treatment applies to.
`Status`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Required. The status of the tax treatment record. Valid values are: • `Active` • `Draft` • `Inactive`

Field	Details
`TaxCode`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The reference code that's used when an external tax engine calculates tax.
`TaxEngineId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ID of the tax engine record that's related to the tax treatment. When tax is calculated for an order item, the tax engine that's related to order item's tax treatment is used. If the tax treatment's `IsTaxable` field value is `True`, the treatment requires a tax engine. This field is a relationship field. **Relationship Name** TaxEngine **Refers To** TaxEngine
`TaxPolicyId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ID of the tax treatment's parent tax policy. This field is a relationship field. **Relationship Name** TaxPolicy **Refers To** TaxPolicy

TransactionJournal

Represents details about billing transactions that are recorded for general ledger accounting purposes. This object is available in API version 63.0 and later.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `undelete()`, `update()`, `upsert()`

Special Access Rules

You need Revenue Cloud Billing and the Accounts Receivables Admin permission set to access this object.

Fields

Field	Details
`Credit`	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The transaction record amount when a credit general ledger account is specified.
`CreditGeneralLedgerAccountId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The ID of the general ledger account for a debit transaction that's related to the transaction journal. This field is a relationship field. **Relationship Name** CreditGeneralLedgerAccount **Refers To** GeneralLedgerAccount
`Debit`	**Type** currency **Properties** Create, Filter, Nillable, Sort, Update **Description** The transaction record amount when a debit general ledger account is specified.
`DebitGeneralLedgerAccountId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The ID of the General Ledger (GL) Treatment based on which transaction journals must be generated for the billing transaction. This field is a relationship field. **Relationship Name** DebitGeneralLedgerAccount **Refers To** GeneralLedgerAccount
`GeneralLedgerAcctAsgntRuleId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The general ledger account assignment rule used to assign general ledger accounts to transaction journals that’s created for billing transactions. This field is a relationship field. **Relationship Name** GeneralLedgerAcctAsgntRule **Refers To** GeneralLedgerAcctAsgntRule
`LegalEntityId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The legal entity that's related to the transaction journal. This field is a relationship field. **Relationship Name** LegalEntity **Refers To** LegalEntity
`ReferenceTransactionRecordId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The transaction record for which the transaction journal is created. This field is a polymorphic relationship field.

Field	Details
	Relationship Name ReferenceTransactionRecord **Refers To** CreditMemo, CreditMemoInvApplication, CreditMemoLine, CreditMemoLineInvoiceLine, CreditMemoLineTax, Invoice, InvoiceLine, InvoiceLineTax, Payment, PaymentLineInvoice, PaymentLineInvoiceLine, Refund
`TransactionType`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The transaction type that's related to the transaction journal. Valid values are: • `CreditMemoLine` • `CreditMemoLineTax` • `InvoiceLine` • `InvoiceLineTax` • `Payment` • `Refund`
`UniqueIdentifier`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** An auto-generated identifier for the transaction journal when the usage type is Billing. The identifier is a combination of the reference transaction record ID, the general ledger account assignment rule ID, and other internal-only fields.
`UsageType`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort **Description** The type of usage. Valid value is: • `Billing`

Billing Platform Events

Salesforce publishes standard platform events in response to an action that occurred in the org or to report errors. For example, the InvoiceProcessedEvent platform event sends notification to the customer when the billing invoice activity is complete. You can subscribe to a standard platform event by using the subscription mechanism that the event supports.

BillingScheduleCreatedEvent
Notifies subscribers when the `/commerce/invoicing/billing-schedules/actions/create` request is complete. This object is available in API version 63.0 and later.

CreditInvoiceProcessedEvent
Represents the notification to the customers after the process initiated by the `/commerce/invoicing/invoices/{invoiceId}/actions/credit` request is complete. This object is available in API version 62.0 and later.

CreditMemoProcessedEvent
Represents the notification to the customers after the process initiated by the `/commerce/invoicing/credit-memos` request is complete. This object is available in API version 62.0 and later.

InvoiceProcessedEvent
Represents the notification to the customers after the process started by the `/commerce/billing/invoices` request is complete. The process groups billing schedules by grouping keys and creates one invoice per grouping key. The `InvoiceProcessedEvent` platform event is a top-level object that contains a list of `InvoiceProcessedDetailEvents` platform events, where each detail event represents an attempt to create one invoice. This object is available in API version 62.0 and later.

NegInvcLineProcessedEvent
Represents the notification to the customers when a negative invoice line is converted to a credit memo This object is available in API version 62.0 and later.

VoidInvoiceProcessedEvent
Represents the notification to the customers after the process started by the `/commerce/invoicing/invoices/{invoiceId}/actions/void` request is complete. The request attempts to void an invoice by crediting an invoice and changing its status to `Voided`, which prevents further changes. This object is available in API version 62.0 and later.

BillingScheduleCreatedEvent

Notifies subscribers when the `/commerce/invoicing/billing-schedules/actions/create` request is complete. This object is available in API version 63.0 and later.

Supported Calls

`describeSObjects()`

Supported Subscribers

Subscriber	Supported?
Apex Triggers	✔

Subscriber	Supported?
Flows	✔
Processes	✔
Pub/Sub API	✔
Streaming API (CometD)	✔

Subscription Channel

`/event/billingschedulecreatedevent`

Event Delivery Allocation Enforced

No

Special Access Rules

This object is available when Billing is enabled.

Fields

Field	Details
`BillingScheduleCreatedEventDetail`	**Type** BillSchdCreatedEventDetail on page 1458 **Properties** Nillable **Description** One `BillingScheduleCreatedEventDetail` entry is created for each order item in the `BillingScheduleCreatedEvent` request. One `BillSchdCreatedEventDetail` entry is created for each error that occurred.
`CorrelationIdentifier`	**Type** string **Properties** Nillable **Description** Reserved for future use.
`EventUuid`	**Type** string **Properties** Nillable

Field	Details
	Description A universally unique identifier (UUID) that identifies a platform event message.
`ReplayId`	**Type** string **Properties** Nillable **Description** Represents an ID value that is populated by the system and refers to the position of the event in the event stream. Replay ID values aren't guaranteed to be contiguous for consecutive events. A subscriber can store a replay ID value and use it on resubscription to retrieve missed events that are within the retention window.
`RequestIdentifier`	**Type** string **Properties** Nillable **Description** ID returned in the `CreateBillingScheduleFromOrderItem` response. Use this ID to identify the `BillingScheduleCreatedEvent` for a specific request.

BillSchdCreatedEventDetail

Contains details about each order item in the `/commerce/invoicing/billing-schedules/actions/create` request and any errors that occurred while processing the request. This object is included in an `BillingScheduleCreatedEvent` message. You can't subscribe to the `BillSchdCreatedEventDetail` platform event directly. This object is available in API version 63.0 and later.

BillSchdCreatedEventDetail

Contains details about each order item in the `/commerce/invoicing/billing-schedules/actions/create` request and any errors that occurred while processing the request. This object is included in an `BillingScheduleCreatedEvent` message. You can't subscribe to the `BillSchdCreatedEventDetail` platform event directly. This object is available in API version 63.0 and later.

Supported Calls

`describeSObjects()`

Special Access Rules

This object is available when Billing is enabled in your org.

Fields

Field	Details
`BillingScheduleId`	**Type** reference **Properties** Nillable **Description** If the request was successful, this field contains the ID of the billing schedule for the order item. This field is a relationship field. **Relationship Name** BillingSchedule **Refers To** BillingSchedule
`ErrorCode`	**Type** string **Properties** Nillable **Description** If the request wasn't successful, this field contains the error code.
`ErrorMessage`	**Type** string **Properties** Nillable **Description** If the request wasn't successful, this field contains the error message.
`EventUuid`	**Type** string **Properties** Nillable **Description** A universally unique identifier (UUID) that identifies a platform event message.
`IsSuccess`	**Type** boolean **Properties** Defaulted on create

Field	Details
	Description Indicates whether the request to create a billing schedule for the order item is successful (`true`) or not (`false`). The default value is `false`.
`OrderItemId`	**Type** reference **Properties** **Description** The ID of the order item used in the `/actions/standardCreateBillingScheduleFromOrderItem` REST request. This field is a relationship field. **Relationship Name** OrderItem **Refers To** OrderItem

CreditInvoiceProcessedEvent

Represents the notification to the customers after the process initiated by the `/commerce/invoicing/invoices/{invoiceId}/actions/credit` request is complete. This object is available in API version 62.0 and later.

Supported Calls

`describeSObjects()`

Supported Subscribers

Subscriber	Supported?
Apex Triggers	✔
Flows	✔
Processes	✔
Pub/Sub API	✔
Streaming API (CometD)	✔

Subscription Channel

`/event/CreditInvoiceProcessedEvent`

Event Delivery Allocation Enforced

No

Special Access Rules

This object is available when Billing is enabled.

Fields

Field	Details
`CorrelationIdentifier`	**Type** string **Properties** Nillable **Description** Reserved for future use.
`CrMemoProcessErrDtlEvents`	**Type** CrMemoProcessErrDtlEvent[] **Properties** Nillable **Description** A compilation of error messages and error codes for a failed request. See the `ErrorDetails` field for error messages and error codes.
`CreditMemoId`	**Type** reference **Properties** Nillable **Description** The credit memo created as the result of a successful request. This field is a relationship field. **Relationship Name** CreditMemo **Refers To** CreditMemo
`ErrorDetails`	**Type** string **Properties** Nillable

Field	Details
	Description If the request fails, this field shows error messages, error codes, and the ID of the record on which the errors occurred.
`EventUuid`	**Type** string **Properties** Nillable **Description** A universally unique identifier (UUID) that identifies a platform event message.
`InvoiceId`	**Type** reference **Properties** Nillable **Description** The invoice credited as the result of a successful request. This field is a relationship field. **Relationship Name** Invoice **Refers To** Invoice
`IsSuccess`	**Type** boolean **Properties** Defaulted on create **Description** Required. Indicates whether the request was successful (`true`) or not (`false`). The default value is `false`.
`ReplayId`	**Type** string **Properties** Nillable **Description** An ID value that's populated by the system and refers to the position of the event in the event stream. Replay ID values aren't guaranteed to be contiguous for consecutive events. A subscriber can store a replay ID value and use it on resubscription to retrieve missed events that are within the retention window.

Field	Details
RequestIdentifier	**Type** string **Properties** Nillable **Description** The unique ID returned in the response. Use this ID to identify the event for a specific request.

CrMemoProcessErrDtlEvent

Represents the information about errors that occurred while creating or applying a credit memo as part of a request. This object is included in a `CreditInvoiceProcessedEvent`, `CreditMemoProcessedEvent`, `NegInvcLineProcessedEvent`, or `VoidInvoiceProcessedEvent` platform event message. You can't subscribe to `CrMemoProcessErrDtlEvent` platform event directly. This object is available in API version 62.0 and later.

CrMemoProcessErrDtlEvent

Represents the information about errors that occurred while creating or applying a credit memo as part of a request. This object is included in a `CreditInvoiceProcessedEvent`, `CreditMemoProcessedEvent`, `NegInvcLineProcessedEvent`, or `VoidInvoiceProcessedEvent` platform event message. You can't subscribe to `CrMemoProcessErrDtlEvent` platform event directly. This object is available in API version 62.0 and later.

Supported Calls

`describeSObjects()`

Special Access Rules

This object is available when Billing is enabled.

Fields

Field	Details
ErrorCode	**Type** string **Properties** Nillable **Description** Reference code for the type of error that occurred.
ErrorMessage	**Type** string **Properties** Nillable

Field	Details
	Description Information about the error that occurred during processing.
`ErrorSourceId`	**Type** reference **Properties** Nillable **Description** The ID of the record on which the error occurred during the credit memo creation process and the application process. This field is a polymorphic relationship field. **Relationship Name** ErrorSource **Refers To** CreditMemo, CreditMemoLine, Invoice, InvoiceLine
`EventUuid`	**Type** string **Properties** Nillable **Description** A universally unique identifier (UUID) that identifies a platform event message.

CreditMemoProcessedEvent

Represents the notification to the customers after the process initiated by the `/commerce/invoicing/credit-memos` request is complete. This object is available in API version 62.0 and later.

Supported Calls

`describeSObjects()`

Supported Subscribers

Subscriber	Supported?
Apex Triggers	✓
Flows	✓
Processes	✓
Pub/Sub API	✓

Subscriber	Supported?
Streaming API (CometD)	✔

Subscription Channel

`/event/CreditMemoProcessedEvent`

Event Delivery Allocation Enforced

No

Special Access Rules

This object is available when Billing is enabled.

Fields

Field	Details
`CorrelationIdentifier`	**Type** string **Properties** Nillable **Description** Reserved for future use.
`CrMemoProcessErrDtlEvents`	**Type** CrMemoProcessErrDtlEvent[] **Properties** Nillable **Description** A compilation of error messages and error codes for a failed request. See the `ErrorDetails` field for error messages and error codes.
`CreditMemoId`	**Type** reference **Properties** Nillable **Description** The credit memo created as the result of a successful request. This field is a relationship field. **Relationship Name** CreditMemo

Field	Details
	Refers To CreditMemo
`ErrorDetails`	**Type** string **Properties** Nillable **Description** If the request fails, this field shows error messages, error codes, and the ID of the record on which the errors occurred.
`EventUuid`	**Type** string **Properties** Nillable **Description** A universally unique identifier (UUID) that identifies a platform event message.
`IsSuccess`	**Type** boolean **Properties** Defaulted on create **Description** Required. Indicates whether the request was successful (`true`) or not (`false`). The default value is `false`.
`ReplayId`	**Type** string **Properties** Nillable **Description** Represents an ID value that's populated by the system and refers to the position of the event in the event stream. Replay ID values aren't guaranteed to be contiguous for consecutive events. A subscriber can store a replay ID value and use it on resubscription to retrieve missed events that are within the retention window.
`RequestIdentifier`	**Type** string **Properties** Nillable **Description** The unique ID returned in the response. Use this ID to identify the event for a specific request.

InvoiceProcessedEvent

Represents the notification to the customers after the process started by the `/commerce/billing/invoices` request is complete. The process groups billing schedules by grouping keys and creates one invoice per grouping key. The `InvoiceProcessedEvent` platform event is a top-level object that contains a list of `InvoiceProcessedDetailEvents` platform events, where each detail event represents an attempt to create one invoice. This object is available in API version 62.0 and later.

Supported Calls

`describeSObjects()`

Supported Subscribers

Subscriber	Supported?
Apex Triggers	✔
Flows	✔
Processes	✔
Pub/Sub API	✔
Streaming API (CometD)	✔

Subscription Channel

`/event/InvoiceProcessedEvent`

Event Delivery Allocation Enforced

No

Special Access Rules

This object is available when Billing is enabled.

Fields

Field	Details
`CorrelationIdentifier`	**Type** string **Properties** Nillable **Description** Reserved for future use.

Field	Details
`EventUuid`	**Type** string **Properties** Nillable **Description** A universally unique identifier (UUID) that identifies a platform event message.
`InvoiceErrorDetailEvent`	**Type** InvoiceErrorDetailEvent[] **Properties** Nillable **Description** Information about errors that occurred during processing.
`InvoiceProcessedDetailEvents`	**Type** InvoiceProcessedDetailEvent[] **Properties** Nillable **Description** A list of `InvoiceProcessedDetailEvent` records. Each record contains information about an attempt to create an invoice from one or more billing schedules that share a grouping key.
`IsSuccess`	**Type** boolean **Properties** Defaulted on create **Description** Required. Indicates whether the Create Order from Invoice action was successful (`true`) or not (`false`). The default value is `false`.
`ReplayId`	**Type** string **Properties** Nillable **Description** An identification (ID) value that's populated by the system and refers to the position of the event in the event stream. Replay ID values aren't guaranteed to be contiguous for consecutive events. A user can store a replay ID value and use it on resubscription to retrieve missed events that are within the retention window.

Field	Details
RequestIdentifier	**Type** string **Properties** Nillable **Description** A unique ID returned in the /commerce/billing/invoices response. Use this ID to identify the event for a specific request.

InvoiceErrorDetailEvent

Represents information about the errors that occurred during the processing of a /commerce/billing/invoices request. This object is included in an InvoiceProcessedEvent platform event message. You can't subscribe to InvoiceProcessedEvent platform event directly. This object is available in API version 62.0 and later.

InvoiceProcessedDetailEvent

Represents the notification to customers regarding the results of an attempt to create an invoice from billing schedules as part of /commerce/billing/invoices request. The InvoiceProcessedDetailEvent platform event contains the results of an attempt to create an invoice from one or more billing schedules that share a grouping key. Each InvoiceProcessedDetailEventplatform event for an action is grouped within the parent InvoiceProcessedDetailEvent platform event. This object is available in API version 62.0 and later.

InvoiceErrorDetailEvent

Represents information about the errors that occurred during the processing of a /commerce/billing/invoices request. This object is included in an InvoiceProcessedEvent platform event message. You can't subscribe to InvoiceProcessedEvent platform event directly. This object is available in API version 62.0 and later.

Supported Calls

describeSObjects()

Special Access Rules

This object is available when Billing is enabled.

Fields

Field	Details
ErrorCode	**Type** string **Properties** None **Description** Reference code for the type of error that occurred.

Field	Details
`ErrorMessage`	**Type** string **Properties** None **Description** Information about the error that occurred during processing.
`ErrorSourceId`	**Type** reference **Properties** Nillable **Description** The ID of the record where the error occurred. This record can be an invoice or a billing schedule. This field is a polymorphic relationship field. **Relationship Name** ErrorSource **Refers To** BillingSchedule, Invoice
`EventUuid`	**Type** string **Properties** Nillable **Description** A universally unique identifier (UUID) that identifies a platform event message.

InvoiceProcessedDetailEvent

Represents the notification to customers regarding the results of an attempt to create an invoice from billing schedules as part of `/commerce/billing/invoices` request. The `InvoiceProcessedDetailEvent` platform event contains the results of an attempt to create an invoice from one or more billing schedules that share a grouping key. Each `InvoiceProcessedDetailEvent`platform event for an action is grouped within the parent `InvoiceProcessedDetailEvent` platform event. This object is available in API version 62.0 and later.

Supported Calls

`describeSObjects()`

Supported Subscribers

Subscriber	Supported?
Apex Triggers	✔
Flows	✔
Processes	✔
Pub/Sub API	✔
Streaming API (CometD)	✔

Subscription Channel

`/event/InvoiceProcessedEvent`

Event Delivery Allocation Enforced

No

Special Access Rules

This object is available when Billing is enabled.

Fields

Field	Details
`EventUuid`	**Type** string **Properties** Nillable **Description** A universally unique identifier (UUID) that identifies a platform event message.
`InvoiceErrorDetailEvents`	**Type** InvoiceErrorDetailEvent[] **Properties** Nillable **Description** A list of errors that occurred while attempting to create the invoice.
`InvoiceId`	**Type** reference **Properties** Nillable

Field	Details
	Description The ID of the new invoice. This field is a relationship field. **Relationship Name** Invoice **Refers To** Invoice
`InvoiceStatus`	**Type** string **Properties** Nillable **Description** The value of the `Status` field on the invoice.
`IsSuccess`	**Type** boolean **Properties** Defaulted on create **Description** Required. Indicates whether the invoice creation attempt was successful (`true`) or not (`false`). The default value is `false`.

NegInvcLineProcessedEvent

Represents the notification to the customers when a negative invoice line is converted to a credit memo This object is available in API version 62.0 and later.

Supported Calls

`describeSObjects()`

Supported Subscribers

Subscriber	Supported?
Apex Triggers	✓
Flows	✓
Processes	✓

Subscriber	Supported?
Pub/Sub API	✔
Streaming API (CometD)	✔

Subscription Channel

`/event/NegInvcLineProcessedEvent`

Event Delivery Allocation Enforced

No

Special Access Rules

This object is available when Billing is enabled.

Fields

Field	Details
`CorrelationIdentifier`	**Type** string **Properties** Nillable **Description** Reserved for future use.
`CrMemoProcessErrDtlEvents`	**Type** CrMemoProcessErrDtlEvent **Properties** Nillable **Description** A compilation of error messages and error codes for a failed request. See the `ErrorDetails` field for error messages and error codes.
`CreditMemoId`	**Type** reference **Properties** Nillable **Description** The ID of the credit memo created as a result of the successful conversion of a negative invoice line. This field is a relationship field.

Field	Details
	Relationship Name CreditMemo **Refers To** CreditMemo
`ErrorDetails`	**Type** string **Properties** Nillable **Description** If the request fails, this field shows error messages, error codes, and the ID of the record on which the errors occurred.
`EventUuid`	**Type** string **Properties** Nillable **Description** A universally unique identifier (UUID) that identifies a platform event message.
`InvoiceId`	**Type** reference **Properties** Nillable **Description** The ID of the invoice that this event is in reference to. This field is a relationship field. **Relationship Name** Invoice **Refers To** Invoice
`IsAutomatedNegativeInvoiceLineConversion`	**Type** boolean **Properties** Defaulted on create **Description** Indicates whether this event is generated either by an automated process to convert negative invoice lines to credit memos (`true`) or by a manual process (`false`). The default value is `false`.

Field	Details
`IsSuccess`	**Type** boolean **Properties** Defaulted on create **Description** Required. Indicates whether the negative invoice lines were converted successfully to credit memos (`true`) or not (`false`). The default value is `false`.
`ReplayId`	**Type** string **Properties** Nillable **Description** Represents an ID value that's populated by the system and refers to the position of the event in the event stream. Replay ID values aren't guaranteed to be contiguous for consecutive events. A subscriber can store a replay ID value and use it on resubscription to retrieve missed events that are within the retention window.
`RequestIdentifier`	**Type** string **Properties** Nillable **Description** The unique identifier of the request. This field is always empty.

VoidInvoiceProcessedEvent

Represents the notification to the customers after the process started by the `/commerce/invoicing/invoices/{invoiceId}/actions/void` request is complete. The request attempts to void an invoice by crediting an invoice and changing its status to `Voided`, which prevents further changes. This object is available in API version 62.0 and later.

Supported Calls

`describeSObjects()`

Supported Subscribers

Subscriber	Supported?
Apex Triggers	✔

Subscriber	Supported?
Flows	✔
Processes	✔
Pub/Sub API	✔
Streaming API (CometD)	✔

Subscription Channel

`/event/VoidInvoiceProcessedEvent`

Event Delivery Allocation Enforced

No

Special Access Rules

This object is available when Billing is enabled.

Fields

Field	Details
`CorrelationIdentifier`	**Type** string **Properties** Nillable **Description** Reserved for future use.
`CrMemoProcessErrDtlEvents`	**Type** CrMemoProcessErrDtlEvent[] **Properties** Nillable **Description** A compilation of error messages and error codes for a failed request. See the `ErrorDetails` field for error messages and error codes.
`CreditMemoId`	**Type** reference **Properties** Nillable

Field	Details
	Description The ID of the credit memo created to void the invoice as the result of a successful request. This field is a relationship field. **Relationship Name** CreditMemo **Refers To** CreditMemo
`ErrorDetails`	**Type** string **Properties** Nillable **Description** If the request fails, this field shows error messages, error codes, and the ID of the record on which the errors occurred.
`EventUuid`	**Type** string **Properties** Nillable **Description** A universally unique identifier (UUID) that identifies a platform event message.
`InvoiceId`	**Type** reference **Properties** Nillable **Description** The invoice that was voided as the result of a successful request. This field is a relationship field. **Relationship Name** Invoice **Refers To** Invoice
`IsSuccess`	**Type** boolean **Properties** Defaulted on create **Description** Required. Indicates whether the request was successful (`true`) or not (`false`).

Field	Details
	The default value is `false`.
`ReplayId`	**Type** string **Properties** Nillable **Description** Represents an ID value that's populated by the system and refers to the position of the event in the event stream. Replay ID values aren't guaranteed to be contiguous for consecutive events. A subscriber can store a replay ID value and use it on resubscription to retrieve missed events that are within the retention window.
`RequestIdentifier`	**Type** string **Properties** Nillable **Description** The unique ID returned in the `/commerce/billing/invoices/{invoiceId}/actions/void` response. Use this ID to identify the event for a specific request.

Billing Standard Invocable Actions

Learn more about the standard invocable actions available with Billing.

Apply Credit Action
Apply a credit memo or credit memo line to an invoice or invoice line, respectively.

Create Billing Schedules From Billing Transaction Action
Create one or more billing schedules for a specified billing transaction ID.

Generate Invoice Documents Action
Asynchronously generate PDF documents for the invoices associated with an invoice batch run record that are in the `Draft` or `Posted` status.

Post Draft Invoice Action
Update the status of an invoice from `Draft` to `Posted` for a credit memo application.

Post Draft Invoice Batch Run Action
Update the status of a batch of invoices from `Draft` to `Posted` for a credit memo application.

Recover Billing Schedules Action
Recover one or more billing schedules in the `Error` or `Processing` status.

Unapply Credit Action

Unapply a credit memo or credit memo line from an invoice or invoice line, respectively.

SEE ALSO:

Actions Developer Guide: Overview

REST API Developer Guide: Invocable Actions Standard

Apply Credit Action

Apply a credit memo or credit memo line to an invoice or invoice line, respectively.

This action credits the amount of the credit memo or credit memo line to the corresponding invoice or invoice line, reducing both their balances.

This action is available in API version 62.0 and later.

Special Access Rules

The Apply Credit action is available in Enterprise, Developer, and Unlimited Editions where Billing is enabled. To use this action, you need the Credit Memo Operations User permission set.

Supported REST HTTP Methods

URI

`/services/data/v63.0/actions/standard/applyCredit`

Formats

JSON, XML

HTTP Methods

POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`appliedCreditAmount`	**Type** double **Description** Required. Credit amount applied to an invoice or invoice line.
`creditSourceRecordId`	**Type** string

Input	Details
	Description Required. ID of the credit memo or credit memo line that's applied to an invoice or invoice line.
`creditTargetRecordId`	**Type** string **Description** Required. ID of the invoice or invoice line record that the credit is applied to.
`description`	**Type** string **Description** Additional details about the credit memo or credit memo line to be applied to an invoice or invoice line.
`effectiveDate`	**Type** string **Description** Date to use for applying the credit memo to an invoice or invoice line.

Outputs

Output	Details
`recordId`	**Type** string **Description** ID of the credit memo invoice application or credit memo line invoice line record of type `Applied` that the action created.

Example

POST

This example shows a sample request for the Apply Credit action.

```
{
  "inputs": [
    {
      "appliedCreditAmount": 20,
```

```
      "creditTargetRecordId": "3ttDU00000000iZYAQ",
      "creditSourceRecordId": "50gDU000000007NYAQ",
      "description": "Applied credit memo to an invoice",
      "effectiveDate": "2024-08-27"
    }
  ]
}
```

This example shows a sample response for the Apply Credit action.

```
{
  "actionName": "applyCredit",
  "errors": null,
  "invocationId": null,
  "isSuccess": true,
  "outputValues": {
    "recordId": "4sFDU00000000652AA"
  },
  "sortOrder": -1,
  "version": 1
}
```

Create Billing Schedules From Billing Transaction Action

Create one or more billing schedules for a specified billing transaction ID.

This action calls the Context-Aware Billing Schedule API to retrieve the billing transaction items associated with the billing transaction ID. The API generates the corresponding billing schedules for each of the billing transaction items for operations such as transaction modifications, renewals, cancellations, and new sales.

This action is available in API version 62.0 and later.

Special Access Rules

The Create Billing Schedules From Billing Transaction action is available in Enterprise, Unlimited, and Developer Editions where Billing is enabled. To use this action, you need the Create Billing Schedules From Billing Transactions API permission set.

Supported REST HTTP Methods

URI

`/services/data/v63.0/actions/standard/createBillingSchedulesFromBillingTransaction`

Formats

JSON, XML

HTTP Methods

POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`billingTransactionId`	**Type** string **Description** Required. ID of the billing transaction record to create one or more billing schedules for.

Outputs

Output	Details
`requestId`	**Type** string **Description** Universally Unique Identifier (UUID) that's used to track the status of the asynchronous action.
`statusUrl`	**Type** string **Description** URL that's used to check the status of the API request.

Example

POST

This example shows a sample request for the Create Billing Schedules From Billing Transaction action.

```
{
  "inputs": [
    {
      "billingTransactionId": "801xx000003JztvAAC"
    }
  ]
}
```

This example shows a sample response for the Create Billing Schedules From Billing Transaction action.

```
{
  "actionName": "createBillingSchedulesFromBillingTransaction",
  "errors": null,
  "isSuccess": true,
  "outputValues": {
    "requestId": "4sFDU00000000652AA",
   "statusUrl": "/services/data/v62.0/sobjects/AsyncOperationTracker/16Pxx0000004NhAEAU"
```

```
  },
  "sortOrder": -1,
  "version": 1
}
```

SEE ALSO:

Revenue Cloud Developer Guide: Context-Aware Billing Schedule API

Generate Invoice Documents Action

Asynchronously generate PDF documents for the invoices associated with an invoice batch run record that are in the `Draft` or `Posted` status.

This action uses the ID of the invoice batch run record to find the draft or posted invoices from the batch and to generate the PDF documents for its invoices that are in the `Draft` or `Posted` status. This action creates a DocGenerationBatchProcess record that contains the Document Generation Process and Invoice Document records for each of the invoices. This action is available in API version 63.0 and later with Revenue Cloud Billing.

Special Access Rules

The Generate Invoice Documents action is available in Enterprise, Unlimited, and Developer Editions where Revenue Cloud Billing is enabled. The org must have Billing Docgen enabled and an active Invoice Document Template to generate PDF documents. To use this action, you need either the Billing Operations User or the Billing Admin permission set, along with the Docgen Designer and Docgen Designer Standard User permission sets.

Supported REST HTTP Methods

URI

`/services/data/v63.0/actions/standard/generateInvoiceDocuments`

Formats

JSON, XML

HTTP Methods

POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`invoiceBatchRunId`	**Type** string **Description** Required. ID of the invoice batch run record that created the draft or posted invoices.

Outputs

Output	Details
`requestId`	**Type** string **Description** An alphanumeric string to track the status of the document generation request.
`requestStatus`	**Type** boolean **Description** Indicates whether the operation is successful (`true`) or not (`false`).

Example

POST

This example shows a sample request for the Generate Invoice Documents action.

```
{
  "inputs": [
    {
      "invoiceBatchRunId": "5IRSG000001Az014AC"
    }
  ]
}
```

This example shows a sample response for the Generate Invoice Documents action.

```
{
  "actionName": "generateInvoiceDocuments",
  "errors": null,
  "isSuccess": true,
  "outputValues": {
    "requestId": "4sFDU00000000652AA",
    "requestStatus": true
  }
}
```

Post Draft Invoice Action

Update the status of an invoice from `Draft` to `Posted` for a credit memo application.

This action uses the ID of the draft invoice and triggers an asynchronous process to post the invoice. This action is available in API version 62.0 and later.

Special Access Rules

The Post Draft Invoice action is available in Enterprise, Unlimited, and Developer Editions where Billing is enabled. To use this action, you need the Billing Operations User permission set.

Supported REST HTTP Methods

URI

`/services/data/v63.0/actions/standard/postDraftInvoice`

Formats

JSON, XML

HTTP Methods

POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`correlationId`	**Type** string **Description** Splunk correlation ID to track the messages that are related to the request and are logged in Splunk by the different services involved in the request. If the ID isn't specified, the service creates a random Universally Unique Identifier (UUID).
`invoiceId`	**Type** string **Description** Required. ID of the `Draft` invoice to be posted.

Outputs

Output	Details
`requestIdentifier`	**Type** string **Description** UUID that's used to track the status of the asynchronous action.

Output	Details
`statusUrl`	**Type** string **Description** URL that's used to check the status of the API request.

Example

POST

This example shows a sample request for the Post Draft Invoice action.

```
{
  "inputs": [
    {
      "invoiceId": "3ttDU00000000iZYAQ"
    }
  ]
}
```

This example shows a sample response for the Post Draft Invoice action.

```
{
  "actionName": "postDraftInvoice",
  "errors": null,
  "isSuccess": true,
  "outputValues": {
    "requestIdentifier": "4sFDU00000000652AA",
   "statusUrl": "/services/data/v62.0/sobjects/AsyncOperationTracker/16Pxx0000004NhAEAU"

  }
}
```

Post Draft Invoice Batch Run Action

Update the status of a batch of invoices from `Draft` to `Posted` for a credit memo application.

This action uses the ID of the invoice batch run record to find draft invoices from the batch and to post the draft invoices to an invoice record. This action is available in API version 62.0 and later.

Special Access Rules

The Post Draft Invoice Batch Run action is available in Enterprise, Unlimited, and Developer Editions where Billing is enabled. To use this action, you need the Billing Operations User permission set.

Supported REST HTTP Methods

URI

`/services/data/v63.0/actions/standard/postDraftInvoiceBatchRun`

Formats

JSON, XML

HTTP Methods

POST

Authentication

`Authorization: Bearer ***token***`

Inputs

Input	Details
`invoiceBatchRunId`	**Type** string **Description** Required. ID of the invoice batch run record that created the draft invoices.

Outputs

Output	Details
`invBatchDraftToPostedRunId`	**Type** string **Description** ID of the record that's created to track the batch process of posting draft invoices. These draft invoices are associated with the parent invoice batch run record.

Example

POST

This example shows a sample request for the Post Draft Invoice Batch Run action.

```
{
  "inputs": [
    {
      "invoiceBatchRunId": "5IRSG000001Az014AC"
    }
  ]
}
```

This example shows a sample response for the Post Draft Invoice Batch Run action.

```
{
  "actionName": "postDraftInvoiceBatchRun",
  "errors": null,
```

```
  "isSuccess": true,
  "outputValues": {
    "invBatchDraftToPostedRunId": "4sFDU00000000652AA"
  }
}
```

Recover Billing Schedules Action

Recover one or more billing schedules in the `Error` or `Processing` status.

This action uses the ID of the billing schedule record in the `Error` or `Processing` status to retrieve the latest generated invoice. This action also retrieves any other billing schedules in the `Error` or `Processing` status associated with that invoice.

This action is available in API version 62.0 and later.

Special Access Rules

The Recover Billing Schedules action is available in Enterprise, Unlimited, and Developer Editions where Billing is enabled. To use this action, you need the Manage Errors Using Invoice Error Recovery API permission set.

Supported REST HTTP Methods

URI

`/services/data/v63.0/actions/standard/recoverBillingSchedules`

Formats

JSON, XML

HTTP Methods

POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`billingScheduleId`	**Type** string **Description** Required. ID of the billing schedule record in the `Error` or `Processing` status.

Outputs

Output	Details
`successBillingScheduleIds`	**Type** string **Description** Comma-separated list of IDs associated with the parent billing schedule record in the `Error` or `Processing` status.

Example

POST

This example shows a sample request for the Recover Billing Schedules action.

```
{
  "inputs": [
    {
      "billingScheduleId": "801xx000003JztvAAC"
    }
  ]
}
```

This example shows a sample response for the Recover Billing Schedules action.

```
{
  "actionName": "recoverBillingSchedules",
  "errors": null,
  "isSuccess": true,
  "outputValues": {
    "successBillingScheduleIds": ["4sFDU00000000652AA", 16Pxx0000004NhAEAU]
  },
  "sortOrder": -1,
  "version": 1
}
```

Unapply Credit Action

Unapply a credit memo or credit memo line from an invoice or invoice line, respectively.

This action unapplies the credit from an invoice or invoice line. This process involves crediting the applied amount of the credit memo invoice application or credit memo line invoice line record to the related credit memo or credit memo line and invoice or invoice line.

This action is available in API version 62.0 and later.

Special Access Rules

The Apply Credit action is available in Enterprise, Developer, and Unlimited Editions where Billing is enabled. To use this action, you need the Credit Memo Operations User permission set.

Supported REST HTTP Methods

URI

`/services/data/v63.0/actions/standard/unapplyCredit`

Formats

JSON, XML

HTTP Methods

POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`effectiveDate`	**Type** string **Description** Date when the credit is unapplied from an invoice or invoice line.
`description`	**Type** string **Description** Additional details about the credit memo invoice application or credit memo line invoice line record of type `Applied` that's processed.
`recordId`	**Type** string **Description** Required. ID of the credit memo invoice application or credit memo line invoice line record of type `Applied` that's processed to unapply the credit.

Outputs

Output	Details
`recordId`	**Type** string **Description** ID of the credit memo invoice application or credit memo line invoice line record of type `Unapplied` that the action created.

Example

POST

This example shows a sample request for the Unapply Credit action.

```
{
  "inputs": [
    {
      "recordId": "4sFDU000000005g2AA",
      "description": "Unapplied credit memo from an invoice",
      "effectiveDate": "2024-08-27"
    }
  ]
}
```

This example shows a sample response for the Unapply Credit action.

```
  {
    "actionName": "unapplyCredit",
    "errors": null,
    "invocationId": null,
    "isSuccess": true,
    "outputValues": {
      "recordId": "4sFDU00000000602AA"
    },
    "sortOrder": -1,
    "version": 1
  }
```

Billing Business APIs

Use the Billing Business APIs to manage credit application and to handle billing scenarios.

These sections list the available resources.

Credits

Resource	Description
`/commerce/invoicing/credit-memos/`***`creditMemoId`***`/actions/apply` (POST)	Adjust or correct already issued invoices by applying an existing credit memo to an invoice.
`/commerce/invoicing/credit-memo-inv-applications/`***`creditMemoInvApplicationId`***`/actions/unapply` (POST)	Unapply a credit memo from an invoice and return the invoice and the credit memo to their pre-application states.
`/commerce/invoicing/credit-memo-lines/`***`creditMemoLineId`***`/actions/apply` (POST)	Adjust or correct already issued invoices by applying an existing credit memo line to an invoice line.

Resource	Description
/commerce/invoicing/credit-memo-line-invoice-line/***creditMemoLineInvoiceLineId***/actions/unapply (POST)	Unapply a credit memo line from an invoice line and return the invoice line and the credit memo line to their pre-application states.
/commerce/invoicing/credit-memos/actions/generate (POST)	Create a credit memo without applying it to an invoice. You can credit the invoice at a later date.
/commerce/invoicing/invoices/***invoiceId***/actions/void (POST)	Void a posted invoice to rebill the customer, if necessary.
/commerce/invoicing/invoices/***invoiceId***/actions/convert-to-credit on page 1529 (POST)	Convert a list of invoice lines with a negative amount into a posted credit memo. This conversion is applicable for a single invoice at a time.
/commerce/invoicing/invoices/***invoiceId***/actions/credit on page 1507 (POST)	Create a credit memo and apply it to an invoice. The credit memo can fully or partially credit the invoice.

Billing Schedules

Resource	Description
/commerce/invoicing/billing-schedules/actions/create (POST)	Generate billing schedules by using context service.
/commerce/invoicing/invoices/collection/actions/generate (POST)	Create an invoice for an account, order, or a list of billing schedules.
/commerce/invoicing/billing-schedules/collection/actions/recover (POST)	Recover the latest generated invoice associated with the billing schedules in the `Error` or `Processing` status.
/commerce/invoicing/actions/suspend-billing (POST)	Suspend billing for billing schedule groups or an account for a predefined period.
/commerce/invoicing/actions/resume-billing (POST)	Resume billing for billing schedule groups or an account that's currently on hold.

Invoices

Resource	Description
`/commerce/invoicing/invoices/collection/actions/post` (POST)	Update the status of the invoice from Draft to Posted.
/commerce/invoicing/invoice-batch-runs/***invoiceBatchRunId***/actions/draft-to-posted (POST)	Update a batch of invoices from `Draft` to `Posted` status for a credit memo application.
`/commerce/invoicing/invoices/collection/actions/preview` (POST)	Generate preview invoices, which includes the estimated tax amounts, for a billing transaction for the next two billing periods.
`/commerce/invoicing/invoices/collection/actions/ingest` (POST)	Ingest or generate an invoice from an internal or external billing transaction data.
/commerce/billing/invoices/invoice-batch-docgen/***invoiceBatchRunId***/actions/***actionName*** (POST)	Asynchronously generate PDF documents for the invoices that are in the `Draft` or `Posted` status and are associated with an invoice batch run record.
/commerce/billing/invoices/invoice-batch-docgen/***invoiceBatchRunId***/actions/***actionName*** (POST)	Asynchronously regenerate PDF documents for the invoices that are in the `Draft` or `Posted` status and failed in an earlier invoice batch run.

Invoice Scheduler

Resource	Description
`/commerce/invoicing/invoice-schedulers` (POST)	Create or update an invoice scheduler to automatically generate invoices. Use the criteria and filters of the invoice scheduler to set up the invoice run schedules based on your requirements.
/commerce/invoicing/invoice-batch-runs/***invoiceBatchRunId***/actions/recover (POST)	Recover records associated with a failed invoice run. Recovery is required only when billing schedules remain in the

Resource	Description
	`Processing` or `Error` status.

Tax Calculation

Resource	Description
`/commerce/taxes/actions/calculate` (POST)	Calculate tax for a transaction.
`/commerce/invoicing/invoices/collection/actions/calculate-estimated-tax` (POST)	Calculate estimated tax for invoices with invoice lines that have the `TaxProcessingStatus` as either `Pending` or `Estimated`.

Billing Business API Limits
Learn about the default limits on the usage of the Billing business APIs.

Resources
Learn more about the available Billing API resources.

Request Bodies
Learn more about the available request bodies of Billing APIs.

Response Bodies
Learn more about the available response bodies of Billing APIs.

SEE ALSO:

Connect REST API Developer Guide: Introduction

Billing Business API Limits

Learn about the default limits on the usage of the Billing business APIs.

Limit	Value
The number of billing schedules that can be processed by the Create Invoices By Using Billing Schedules API.	200
The number of invoice lines that can be generated by the Create Invoices By Using Billing Schedules API.	200

EDITIONS

Available in: Lightning Experience

Available in: **Enterprise**, **Unlimited**, and **Developer** Editions

Limit	Value
The number of billing schedules that can be processed by the Recover Billing Schedule List API.	200
The number of invoices that can be processed by the Apply Credit Memos API.	300
The number of invoice lines that can be processed by the Apply Credit Memo Lines API.	300
The number of invoices that can be processed by the Create and Apply Credit Memos API.	300
The number of credit memo lines of type Charge that can be created by the Create a Standalone Credit Memo API.	300
The number of invoice lines that can be processed by the Convert Negative Invoice Lines to Credits API.	300
The number of billing transaction items that can be processed by the Create Billing Schedules From Billing Transactions API.	200
The number of reference IDs that can be processed by the Suspend Billing API and the Resume Billing API.	200
The number of records that can be processed by the Invoice Ingestion API.	500
The number of invoice lines that can be processed by the Invoice Preview API.	200

Resources

Learn more about the available Billing API resources.

Batch Invoices Document Generation (POST)

Asynchronously generate PDF documents for the invoices that are in the `Draft` or `Posted` status and are associated with an invoice batch run record.

Batch Invoices Document Generation Retry (POST)

Asynchronously regenerate PDF documents for the invoices that are in the `Draft` or `Posted` status and failed in an earlier invoice batch run.

Batch Invoices Draft to Posted Status (POST)

Update a batch of invoices from `Draft` to `Posted` status for a credit memo application.

Batch Invoice Scheduler (POST, PUT)

Create or update an invoice scheduler to automatically generate invoices. Use the criteria and filters of the invoice scheduler to set up the invoice run schedules based on your requirements.

Billing Schedule Recovery List (POST)

Recover the latest generated invoice associated with the billing schedules in the `Error` or `Processing` status.

Context-Aware Billing Schedule (POST)

Generate billing schedules by using context service.

Credit Memo Create and Apply (POST)

Create a credit memo and apply it to an invoice. The credit memo can fully or partially credit the invoice.

Credit Memo Apply List (POST)

Adjust or correct already issued invoices by applying an existing credit memo to an invoice.

Credit Memo Unapply (POST)

Unapply a credit memo from an invoice and return the invoice and the credit memo to their pre-application states.

Credit Memo Line Apply (POST)

Adjust or correct already issued invoices by applying an existing credit memo line to an invoice line.

Credit Memo Line Unapply (POST)

Unapply a credit memo line from an invoice line and return the invoice line and the credit memo line to their pre-application states.

Invoice Creation (POST)

Create an invoice for an account, order, or a list of billing schedules.

Invoice Draft to Posted Status (POST)

Update the status of the invoice from Draft to Posted.

Invoice Ingestion (POST)

Ingest or generate an invoice from an internal or external billing transaction data.

Invoice Estimated Tax Calculation (POST)

Calculate estimated tax for invoices with invoice lines that have the `TaxProcessingStatus` as either `Pending` or `Estimated`.

Invoice Preview (POST)

Generate preview invoices, which includes the estimated tax amounts, for a billing transaction for the next two billing periods.

Invoice Run Recovery (POST)

Recover records associated with a failed invoice run. Recovery is required only when billing schedules remain in the `Processing` or `Error` status.

Negative Invoice Lines to Credit Conversion (POST)

Convert a list of invoice lines with a negative amount into a posted credit memo. This conversion is applicable for a single invoice at a time.

Posted Invoice Voidance (POST)

Void a posted invoice to rebill the customer, if necessary.

Resume Billing (POST)

Resume billing for billing schedule groups or an account that's currently on hold.

Suspend Billing (POST)

Suspend billing for billing schedule groups or an account for a predefined period.

Standalone Credit Memo (POST)

Create a credit memo without applying it to an invoice. You can credit the invoice at a later date.

Tax Calculation (POST)

Calculate tax for a transaction.

Batch Invoices Document Generation (POST)

Asynchronously generate PDF documents for the invoices that are in the `Draft` or `Posted` status and are associated with an invoice batch run record.

Special Access Rules

This API is available with Revenue Cloud Billing. To use this API, enable Document Generation for Billing. Additionally, you need either the Billing Operations User permission set or the Billing Admin permission set, along with the Docgen Designer permission set and Docgen Designer Standard User permission set.

Resource

```
/commerce/billing/invoices/invoice-batch-docgen/invoiceBatchRunId/actions/actionName
```

- The *`invoiceBatchRunId`* parameter is the ID of the invoice batch run record that created the `Draft` or `Posted` invoices.
- The *`actionName`* parameter is the name of the action you want to perform on the specified invoice batch run record.

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/commerce/billing/invoices/invoice-batch-docgen/5KPxx0025063G3mGAX/actions/run
```

Available version

63.0

HTTP methods

POST

Response body for POST

Batch Invoice Document Generation

Batch Invoices Document Generation Retry (POST)

Asynchronously regenerate PDF documents for the invoices that are in the `Draft` or `Posted` status and failed in an earlier invoice batch run.

Special Access Rules

This API is available with Revenue Cloud Billing. To use this API, enable Document Generation for Billing. Additionally, you need either the Billing Operations User permission set or the Billing Admin permission set, along with the Docgen Designer permission set and Docgen Designer Standard User permission set.

Resource

```
/commerce/billing/invoices/invoice-batch-docgen/invoiceBatchRunId/actions/actionName
```

- The *`invoiceBatchRunId`* parameter is the ID of the invoice batch run record that created the `Draft` or `Posted` invoices.
- The *`actionName`* parameter is the name of the action you want to perform on the specified invoice batch run record. In this case, the action is to retry generating the `Draft` or `Posted` invoices that failed earlier.

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/commerce/billing/invoices/invoice-batch-docgen/5IRxx0000004G3mGAM/actions/retry
```

Available version

63.0

HTTP methods

POST

Response body for POST

Batch Invoice Document Generation

Batch Invoices Draft to Posted Status (POST)

Update a batch of invoices from `Draft` to `Posted` status for a credit memo application.

Special Access Rules

To use this API, you need the Billing Operations User permission set.

Resource

```
/commerce/invoicing/invoice-batch-runs/invoiceBatchRunId/actions/draft-to-posted
```

This API posts the draft invoices and changes the status of the invoices from `Draft` to `Posted`.

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/commerce/invoicing/invoice-batch-runs/5IRN0000000cA94AI/actions/draft-to-posted
```

Available version

62.0

HTTP methods

POST

Path parameter for POST

Name	Type	Description	Required or Optional	Available Version
`invoiceBatchRunId`	String	ID of the invoice batch run record that creates the draft invoices.	Required	62.0

Response body for POST

Invoice Batch Draft To Posted

Batch Invoice Scheduler (POST, PUT)

Create or update an invoice scheduler to automatically generate invoices. Use the criteria and filters of the invoice scheduler to set up the invoice run schedules based on your requirements.

Special Access Rules

You need the Billing Operations User and Data Pipelines Base User permission sets to use this API. See Data Pipelines.

Resource

```
/commerce/invoicing/invoice-schedulers
```

```
/commerce/invoicing/invoice-schedulers/billingBatchSchedulerId
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/commerce/invoicing/invoice-schedulers
```

```
https://yourInstance.salesforce.com/services/data/v63.0/commerce/invoicing/invoice-schedulers/5BSxx0000004TwGGAU
```

Available version

62.0

HTTP methods

POST, PUT

PUT is supported in version 63.0 and later for invoice schedulers with `Draft` or `Inactive` status.

Request body for POST

JSON example

This example shows a sample request to create an invoice scheduler that generates invoices once.

```
{
  "schedulerName": "InvoiceScheduler",
  "startDate": "2024-05-06",
  "invoiceStatus": "POSTED",
  "preferredTime": "00:45",
  "targetDate": "2024-05-22",
  "invoiceDate": "2024-05-22",
  "frequencyCadence": "Once",
  "frequencyCadenceOptions": {},
  "timezone": "Asia/Kolkata",
  "status": "Active",
  "filterCriteria": [
    {
      "operation": "InList",
      "value": "Batch 2,Batch 3,Batch 4",
      "criteriaSequence": 1,
      "objectName": "BillingSchedule",
      "fieldName": "InvoiceRunMatchingValue"
    },
    {
      "operation": "Equals",
      "value": "001xx000003GZG5AAO",
      "criteriaSequence": 2,
      "objectName": "BillingSchedule",
      "fieldName": "BillingAccount"
    },
    {
      "operation": "Equals",
      "value": "0fwxx0000000001AAA",
      "criteriaSequence": 3,
      "objectName": "BillingScheduleGroup",
      "fieldName": "LegalEntity"
    },
    {
      "operation": "InList",
      "value": "OneTime,Recurring",
      "criteriaSequence": 4,
      "objectName": "BillingSchedule",
```

```
      "fieldName": "BillingTermUnit"
    },
    {
      "operation": "Equals",
      "value": "USD",
      "criteriaSequence": 5,
      "objectName": "BillingSchedule",
      "fieldName": "Currency_Iso_code"
    }
  ]
}
```

This example shows a sample request to create an invoice scheduler that generates invoices daily.

```
{
  "schedulerName": "InvoiceScheduler",
  "startDate": "2024-05-06",
  "endDate": "2026-05-06",
  "invoiceStatus": "POSTED",
  "preferredTime": "00:45",
  "targetDateOffset": 0,
  "invoiceDateOffset": 0,
  "isInvoiceDateFromRunDate": true,
  "frequencyCadence": "Daily",
  "frequencyCadenceOptions": {},
  "timezone": "Asia/Kolkata",
  "status": "Active",
  "filterCriteria": [
    {
      "operation": "InList",
      "value": "Batch 2,Batch 3,Batch 4",
      "criteriaSequence": 1,
      "objectName": "BillingSchedule",
      "fieldName": "InvoiceRunMatchingValue"
    },
    {
      "operation": "Equals",
      "value": "001xx000003GZG5AAO",
      "criteriaSequence": 2,
      "objectName": "BillingSchedule",
      "fieldName": "BillingAccount"
    },
    {
      "operation": "Equals",
      "value": "0fwxx0000000001AAA",
      "criteriaSequence": 3,
      "objectName": "BillingScheduleGroup",
      "fieldName": "LegalEntity"
    },
    {
      "operation": "InList",
      "value": "OneTime,Recurring",
      "criteriaSequence": 4,
      "objectName": "BillingSchedule",
      "fieldName": "BillingTermUnit"
```

```
    },
    {
      "operation": "InList",
      "value": "USD",
      "criteriaSequence": 5,
      "objectName": "BillingSchedule",
      "fieldName": "Currency_Iso_code"
    }
  ]
}
```

This example shows a sample request to create an invoice scheduler that generates invoices weekly.

```
{
  "schedulerName": "InvoiceScheduler",
  "startDate": "2024-05-06",
  "endDate": "2026-05-06",
  "invoiceStatus": "POSTED",
  "preferredTime": "00:45",
  "targetDateOffset": 0,
  "invoiceDateOffset": 0,
  "isInvoiceDateFromRunDate": false,
  "frequencyCadence": "Weekly",
  "frequencyCadenceOptions": {
    "recursOnDay": "Sunday"
  },
  "timezone": "Asia/Kolkata",
  "status": "Active",
  "filterCriteria": [
    {
      "operation": "InList",
      "value": "Batch 2,Batch 3,Batch 4",
      "criteriaSequence": 1,
      "objectName": "BillingSchedule",
      "fieldName": "InvoiceRunMatchingValue"
    },
    {
      "operation": "Equals",
      "value": "001xx000003GZG5AAO",
      "criteriaSequence": 2,
      "objectName": "BillingSchedule",
      "fieldName": "BillingAccount"
    },
    {
      "operation": "Equals",
      "value": "0fwxx0000000001AAA",
      "criteriaSequence": 3,
      "objectName": "BillingScheduleGroup",
      "fieldName": "LegalEntity"
    },
    {
      "operation": "InList",
      "value": "OneTime,Recurring",
      "criteriaSequence": 4,
      "objectName": "BillingSchedule",
```

```
      "fieldName": "BillingTermUnit"
    },
    {
      "operation": "Equals",
      "value": "USD",
      "criteriaSequence": 5,
      "objectName": "BillingSchedule",
      "fieldName": "Currency_Iso_code"
    }
  ]
}
```

This example shows a sample request to create an invoice scheduler that generates invoices monthly on a specific date.

```
{
  "schedulerName": "InvoiceScheduler",
  "startDate": "2024-05-06",
  "endDate": "2026-05-06",
  "invoiceStatus": "POSTED",
  "preferredTime": "00:45",
  "targetDateOffset": 0,
  "invoiceDateOffset": 0,
  "isInvoiceDateFromRunDate": false,
  "frequencyCadence": "Monthly",
  "frequencyCadenceOptions": {
    "recurringSubType": "SpecificDate",
    "recursOnDate": "L-1",
    "shouldExcludeWkendAndHldy": true
  },
  "timezone": "Asia/Kolkata",
  "status": "Active",
  "filterCriteria": [
    {
      "operation": "InList",
      "value": "Batch 2,Batch 3,Batch 4",
      "criteriaSequence": 1,
      "objectName": "BillingSchedule",
      "fieldName": "InvoiceRunMatchingValue"
    },
    {
      "operation": "Equals",
      "value": "001xx000003GZG5AAO",
      "criteriaSequence": 2,
      "objectName": "BillingSchedule",
      "fieldName": "BillingAccount"
    },
    {
      "operation": "Equals",
      "value": "0fwxx0000000001AAA",
      "criteriaSequence": 3,
      "objectName": "BillingScheduleGroup",
      "fieldName": "LegalEntity"
    },
    {
      "operation": "InList",
```

```
      "value": "OneTime,Recurring",
      "criteriaSequence": 4,
      "objectName": "BillingSchedule",
      "fieldName": "BillingTermUnit"
    },
    {
      "operation": "Equals",
      "value": "USD",
      "criteriaSequence": 5,
      "objectName": "BillingSchedule",
      "fieldName": "Currency_Iso_code"
    }
  ]
}
```

This example shows a sample request to create an invoice scheduler that runs immediately.

```
{
    "schedulerName": "InvoiceScheduler",
    "status": "Draft",
    "invoiceStatus": "Posted",
    "frequencyCadenceOptions": {
        "shouldStartRunImmediately": true
    },
    "frequencyCadence": "Once",
    "targetDate": "2024-08-28",
    "invoiceDate": "2024-08-28",
    "filterCriteria": [
        {
            "operation": "InList",
            "value": "Batch 2,Batch 3,Batch 4",
            "criteriaSequence": 1,
            "objectName": "BillingSchedule",
            "fieldName": "InvoiceRunMatchingValue"
        },
        {
            "operation": "Equals",
            "value": "001xx000003GZG5AAO",
            "criteriaSequence": 2,
            "objectName": "BillingSchedule",
            "fieldName": "BillingAccount"
        },
        {
            "operation": "Equals",
            "value": "0fwxx0000000001AAA",
            "criteriaSequence": 3,
            "objectName": "BillingScheduleGroup",
            "fieldName": "LegalEntity"
        },
        {
            "operation": "InList",
            "value": "OneTime,Recurring",
            "criteriaSequence": 4,
            "objectName": "BillingSchedule",
            "fieldName": "BillingTermUnit"
```

```
        },
        {
            "operation": "Equals",
            "value": "USD",
            "criteriaSequence": 5,
            "objectName": "BillingSchedule",
            "fieldName": "Currency_Iso_code"
        }
    ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`endDate`	String	End date of the invoice scheduler.	Optional	63.0
`filterCriteria`	Batch Invoice Filter Criteria Input[]	List of line items of the filter criteria.	Optional	62.0
`frequencyCadence`	String	Frequency to run the invoice scheduler. Valid values are: • `Once` • `Daily` • `Weekly` • `Monthly`	Required	62.0
`frequencyCadenceOptions`	Frequency Cadence Options	Frequency cadence options for the invoice scheduler.	Required	62.0
`invoiceDate`	String	Date shown on the invoice. This date is also used for tax calculations.	Required if the `frequencyCadence` property is set to `Once`.	62.0
`invoiceDateOffset`	Integer	Offset applied to the target date, which is the number of days added to or subtracted from the invoice date, to calculate the updated invoice date.	Required if the `frequencyCadence` property is set to `Daily`, `Weekly`, or `Monthly`.	62.0
`invoiceStatus`	String	Status of the invoice that specifies the expected invoice status from an invoice batch run. Valid values are: • `Draft` • `Posted`	Required	62.0

Name	Type	Description	Required or Optional	Available Version
`isInvoiceDateFromRunDate`	Boolean	Indicates whether the invoice date is applicable from the date when the invoice scheduler is run (`true`) or not (`false`).	Optional	63.0
`preferredTime`	String	Preferred time for the invoice batch run.	Required	62.0
`schedulerName`	String	Name of the invoice scheduler, which must be unique in your org.	Required	62.0
`startDate`	String	Start date of the invoice scheduler.	Required	62.0
`status`	String	Status of the invoice scheduler. Valid values are: • `Draft` • `Active` • `Inactive`	Required	62.0
`targetDate`	String	Target date of the invoice batch run. Billing schedules having the next billing date before this date are picked up for invoicing. The target date must be less than or equal to the maximum allowed target date for the org.	Required if the `frequencyCadence` property is set to `Once`.	62.0
`targetDateOffset`	Integer	Target date offset applied to the next invoice run date to calculate the target date. The offset is the number of days added to or subtracted from the next billing date.	Required if the `frequencyCadence` property is set to `Daily`, `Weekly`, or `Monthly`.	62.0
`timezone`	String	Time zone that's applicable for the invoice scheduler.	Optional	62.0

Response body for POST

Batch Invoice Scheduler

Billing Schedule Recovery List (POST)

Recover the latest generated invoice associated with the billing schedules in the `Error` or `Processing` status.

Billing schedules include critical details such as the amount to be billed, next billing date, and status. An invoice can be associated with one or more billing schedules. When an invoice is generated or posted, the billing schedules are updated to reflect the accurate state

of the invoice. The billing schedules associated with an invoice are marked in the `Error` status if any of the invoicing processes have errors. Use this API to recover the invoice associated with the billing schedules in the `Error` or `Processing` status.

Special Access Rules

You need the Manage Errors Using Invoice Error Recovery API permission set to use this API.

Resource

```
/commerce/invoicing/billing-schedules/collection/actions/recover
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/commerce/invoicing/billing-schedules/collection/actions/recover
```

Available version

62.0

HTTP methods

POST

Request body for POST

JSON example

```
{
  "billingScheduleIds": ["44bDU00000000XXYAY"]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`billingScheduleIds`	String[]	IDs of the billing schedules to recover the invoice for. You can recover one billing schedule per API request.	Required	62.0

Response body for POST

Billing Schedule Recovery List

Context-Aware Billing Schedule (POST)

Generate billing schedules by using context service.

Special Access Rules

The org must have the standard billing context definition with the target mapping. This context definition is available in orgs with Billing enabled. Additionally, you need the Create Billing Schedules From Billing Transactions API permission set and the Context Service Runtime permission set to use this API.

Resource

```
/commerce/invoicing/billing-schedules/actions/create
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/commerce/invoicing/billing-schedules/actions/create
```

Available version

62.0

HTTP methods

POST

Request body for POST

JSON example

```
{
    "billingTransactionIds": [ "801xx000003H1H9AAK"]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
billingTransactionIds	String[]	ID of the billing transaction. This property value is the ID of the order if the source of the billing request is for the Order object. If the order product associated with the specified order ID doesn't have an associated billing treatment ID, the API considers the default billing treatment ID. The generated billing schedule group has the default billing treatment ID. The API supports only one billing transaction ID in the input.	Required	62.0

Response body for POST

Context Aware Billing Schedule

Credit Memo Create and Apply (POST)

Create a credit memo and apply it to an invoice. The credit memo can fully or partially credit the invoice.

Use this API to adjust an outstanding invoice balance or rectify errors in an invoice. In the API request, pass a list of invoice lines to credit. Keep these considerations in mind when you use this API.

- The request must contain at least one invoice line. Each invoice line must have the invoice line's ID, the amount to credit, and any optional tax details. The invoice lines must be a part of the invoice passed in the resource.
- The amount to credit must not exceed the charge or adjustment amount of an individual invoice line.
- The request body's credit amount inclusive of taxes must not exceed the target invoice line's amount inclusive of taxes, except for taxes calculated through an external tax service.
- The request body's total credit amount inclusive of taxes calculated through an external tax service must not exceed the outstanding invoice balance, which is also inclusive of taxes.

This API creates and posts a credit memo. The credit memo has one credit memo line for each invoice line passed in the API request. The invoice balance reduces by a value equal to the credit memo's balance. This API modifies the balance of a posted invoice or invoice line based on the specified credit application level for your org. See Apply Credits to Posted Invoices or Invoice Lines.

Special Access Rules

You need the Credit Memo Operations User permission set to use this API.

Resource

```
/commerce/invoicing/invoices/invoiceId/actions/credit
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/commerce/invoicing/invoices/3ttR00000000NEIAY/actions/credit
```

Available version

62.0

HTTP methods

POST

Path parameter for POST

Name	Type	Description	Required or Optional	Available Version
`invoiceId`	String	ID of the invoice to be credited partially or fully. The status of the invoice must be `Posted`.	Required	62.0

Request body for POST

JSON example

This example shows a sample request with the `Calculate` tax strategy.

```
{
  "type": "POSTED",
  "taxStrategy": "Calculate",
  "taxEffectiveDate": "2024-07-20",
  "effectiveDate": "2024-07-20",
  "description": "Credit Invoice",
  "invoiceLines": [
    {
      "invoiceLineId": "5TVR00000004SiqOBE",
      "amountToCredit": 5
    }
  ]
}
```

This example shows a sample request with the `CopyFromInvoiceLine` tax strategy.

```
{
  "type": "POSTED",
  "taxStrategy": "CopyFromInvoiceLine",
  "effectiveDate": "2020-05-22",
  "description": "Credit Invoice",
```

```
  "invoiceLines": [
    {
      "invoiceLineId": "5TVR00000004SiqOBE",
      "amountToCredit": "5",
      "taxStrategy": "CopyFromInvoiceLine"
    }
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`description`	String	Description for the credit memo to be created.	Optional	62.0
`effectiveDate`	String	Date when the credit memo takes effect.	Optional	62.0
`invoiceLines`	Credit Invoice Line Input[]	List of the invoice lines to be credited. The invoice line IDs must be related to the invoice ID specified in the API request. If invoice lines aren't specified, the API request results in an error.	Required	62.0
`taxEffectiveDate`	String	Date when the tax takes effect to recalculate the taxes.	Optional	62.0
`taxStrategy`	String	Tax strategy to be applied across invoice lines. You can override the tax strategy at the individual invoice line level or at the tax line level. Valid values are: • `Ignore`—Specifies that the creation of tax lines must be ignored. • `CopyFromInvoiceLine`—Specifies that tax values must be copied from the invoice line. • `Calculate`—Specifies that tax must be calculated by using the API.	Required	62.0
`type`	String	Type of credit memo to be created. Valid value is `Posted`.	Optional	62.0

Response body for POST

Revenue Async Line Level

Credit Memo Apply List (POST)

Adjust or correct already issued invoices by applying an existing credit memo to an invoice.

Specify the credit memo ID and the amounts to be applied, with the total of all applied amounts not exceeding the credit memo's balance.

The credit amount for each invoice can't surpass the original charge or adjustment amount, and the overall credit amount must not exceed the invoice's outstanding balance. The exceptions include any taxes calculated by an external service.

For example, your organization sold 10 tablets at US$500 each, totaling $5000, to a vendor who later reported that 6 tablets were defective. Your accounts receivable team creates a $3000 credit memo by using the standalone Credit Memo API. Additionally, the team applies this credit to the original invoice by using the Credit Memo Apply API.

Special Access Rules

You need the Credit Memo Operations User permission set to use this API.

Resource

```
/commerce/invoicing/credit-memos/creditMemoId/actions/apply
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/commerce/invoicing/credit-memos/50gSG0000000XATYA2/actions/apply
```

Available version

62.0

HTTP methods

POST

Path parameter for POST

Name	Type	Description	Required or Optional	Available Version
`creditMemoId`	String	ID of the credit memo record.	Required	62.0

Request body for POST

JSON example

```
{
  "applications": [
    {
      "appliedToId": "3ttxx000000003FAAQ",
      "amount": 10,
      "description": "Apply to invoice for refund",
      "effectiveDate": "2024-07-01"
    },
    {
      "appliedToId": "3ttxx0000000001AAA",
      "amount": 100
    }
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`applications`	Credit Memo Apply Application Input[]	List of one or more applications to apply the credit memo for. Each application represents an invoice that's credited by using the balance of the specified credit memo.	Required	62.0

Response body for POST

Credit Memo Apply List

Credit Memo Unapply (POST)

Unapply a credit memo from an invoice and return the invoice and the credit memo to their pre-application states.

Use this resource if an error occurred when a credit is issued. For example, if an incorrect credit memo is applied to an invoice, or if a credit memo is created for an incorrect amount, use this resource to unapply the credit memo.

Special Access Rules

You need the Credit Memo Operations User permission set to use this API.

Resource

```
/commerce/invoicing/credit-memo-inv-applications/creditMemoInvApplicationId/actions/unapply
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/commerce/invoicing/credit-memo-inv-applications/4sESG000002mBB2AY/actions/unapply
```

Available version

62.0

HTTP methods

POST

Path parameter for POST

Name	Type	Description	Required or Optional	Available Version
`creditMemoInvApplicationId`	String	ID of the credit memo invoice application.	Required	62.0

Request body for POST

JSON example

```
{
  "description": "Unapply credit memo from invoice to revert an error",
```

```
    "effectiveDate": "2024-07-01"
  }
```

Properties

Name	Type	Description	Required or Optional	Available Version
`description`	String	Explanation or reason for unapplying the credit memo.	Optional	62.0
`effectiveDate`	String	Effective date for the credit memo.	Optional	62.0

Response body for POST

Credit Memo Unapply

Credit Memo Line Apply (POST)

Adjust or correct already issued invoices by applying an existing credit memo line to an invoice line.

Special Access Rules

You need the Credit Memo Operations User permission set to use this API.

Resource

```
/commerce/invoicing/credit-memo-lines/creditMemoLineId/actions/apply
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/commerce/invoicing/credit-memo-lines/9xxSG0000000LLHYA2/actions/apply
```

Available version

62.0

HTTP methods

POST

Path parameter for POST

Name	Type	Description	Required or Optional	Available Version
`creditMemoLineId`	String	ID of the credit memo line record.	Required	62.0

Request body for POST

JSON example

```
{
 "applyCreditDetails": [
 {
 "invoiceLineId": "5TVSG0000002ZJR4A2",
 "appliedAmount": 5,
```

```
 "description": "Apply to invoice line 1",
 "effectiveDate": "2024-07-01"
 },
 {
 "invoiceLineId": "5TVSG0000002ZJS4A2",
 "appliedAmount": 10,
 "description": "Apply to invoice line 2",
 "effectiveDate": "2024-07-01"
 }
 ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`applyCreditDetails`	Credit Memo Line Application Input[]	List of one or more applications to apply the credit memo line for. Each application represents an invoice line that's credited by using the balance of the specified credit memo line.	Required	62.0

Response body for POST

Credit Memo Line Applied

Credit Memo Line Unapply (POST)

Unapply a credit memo line from an invoice line and return the invoice line and the credit memo line to their pre-application states.

Special Access Rules

You need the Credit Memo Operations User permission set to use this API.

Resource

```
/commerce/invoicing/credit-memo-line-invoice-line/creditMemoLineInvoiceLineId/actions/unapply
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/commerce/invoicing/credit-memo-line-invoice-line/4sG3G0000002kQ92AI/actions/unapply
```

Available version

62.0

HTTP methods

POST

Path parameter for POST

Name	Type	Description	Required or Optional	Available Version
`creditMemoLineInvoiceLineId`	String	ID of the credit memo line invoice line record.	Required	62.0

Request body for POST

JSON example

```
{
 "description": "Unapply a credit memo line from invoice line 1",
 "effectiveDate": "2024-07-01"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`description`	String	Explanation or reason for unapplying the credit memo line.	Optional	62.0
`effectiveDate`	String	Effective date for the credit memo line.	Optional	62.0

Response body for POST

Credit Memo Line Unapplied

Invoice Creation (POST)

Create an invoice for an account, order, or a list of billing schedules.

This API request creates billing period items for the matching billing schedules. The billing period items are created for the period between the next charge date of the billing schedule and the specified target date. Invoice lines are created by processing these billing period items. These invoice lines are then grouped into invoices based on the defined grouping criteria on the billing schedule.

This API also applies any available credits on an account to settle an invoice and to reduce it's balance. To apply available credits, ensure the **Apply Credits to Posted Invoices** setting from Billing Settings is turned on.

Special Access Rules

You need the Generate Invoices From Billing Schedule API, Billing Operations User, or Billing Customer Service permission set to use this API.

Resource

```
/commerce/invoicing/invoices/collection/actions/generate
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/commerce/invoicing/invoices/collection/actions/generate
```

Available version

62.0

HTTP methods

POST

Request body for POST

Note: The `billingTransactionId` property takes precedence over the `accountId` property followed by the `billingScheduleIds` property when values for these properties are specified in the input request.

JSON example

```
{
  "accountId": "001SG00000mYtRWYA0",
  "action": "Posted",
  "billingScheduleIds": [
    "44bSG000000CVeMYAW"
  ],
  "billingTransactionId": "801SG00000mYtaXYAS",
  "correlationId": null,
  "invoiceDate": "2024-01-12",
  "targetDate": "2024-01-12"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`accountId`	String	ID of the account record to create the invoices for.	Required if the `billingScheduleIds` or `billingTransactionId` property isn't specified.	63.0
`action`	String	Type of invoice to be created. Valid values are: • `Draft` • `Posted`	Required	62.0
`billingScheduleIds`	String[]	List of billing schedule IDs that's used to create the invoices. You can specify a maximum of 200 billing schedules.	Required if the `accountId` or `billingTransactionId` property isn't specified.	62.0
`billingTransactionId`	String	ID of the billing transaction record, which is the order ID, to create the invoices for.	Required if the `accountId` or `billing`	63.0

Name	Type	Description	Required or Optional	Available Version
			`ScheduleIds` property isn't specified.	
`correlationId`	String	Property that's tagged against the published `InvoiceProcessedEvent` event, if specified.	Optional	62.0
`invoiceDate`	String	Stamping date of the invoice in ISO 8601 format.	Required	62.0
`targetDate`	String	Date in ISO 8601 format used to decide the billing periods that are included to create invoices.	Required	62.0

Response body for POST

Revenue Async Response

Invoice Draft to Posted Status (POST)

Update the status of the invoice from Draft to Posted.

This API calls an external tax engine or provides information to your tax adapter implementation to calculate taxes for the draft invoice, post the invoice, and update the related billing schedules and billing periods.

Special Access Rules

You need the Billing Operations User permission set to use this API.

Resource

```
/commerce/invoicing/invoices/collection/actions/post
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/commerce/invoicing/invoices/collection/actions/post
```

Available version

62.0

HTTP methods

POST

Request body for POST

JSON example

```
{
  "invoiceIds": ["3ttxx0000004CIjAAM"]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`correlationId`	String	Splunk correlation ID to track the messages that are related to the request and are logged in Splunk by the different services involved in the request. If the ID isn't specified, the service creates a random Universally Unique Identifier (UUID).	Optional	62.0
`invoiceIds`	String[]	IDs of the invoice records in `Draft` status to be posted. You can post one draft invoice per API request.	Required	62.0

Response body for POST

Revenue Async Response

Invoice Ingestion (POST)

Ingest or generate an invoice from an internal or external billing transaction data.

Create invoices from any external or internal transaction. For example, create invoices by using external orders, policies, or quotes, or internal data such as assets or quotes.

You can also ingest invoices from any external system into Billing, and perform any invoice operations such as issuing credits against an invoice or voiding an invoice.

Keep these considerations in mind when you use this API to ingest a negative invoice line.

- Use the Negative Invoice Lines to Credit Conversion API to create a credit memo against an ingested negative invoice line.
- Use the Apply Credit Action to apply a credit memo to an invoice.

Resource

```
/commerce/invoicing/invoices/collection/actions/ingest
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/commerce/invoicing/invoices/collection/actions/ingest
```

Available version

63.0

HTTP methods

POST

Request body for POST

JSON example

This example shows a sample request to ingest a draft invoice with a tax callout.

```
{
  "invoices": [
    {
      "shouldCalculateTax": true,
      "taxCalculationStatus": "Estimated",
      "graph": {
        "graphId": "CreateInvoice",
        "records": [
          {
            "referenceId": "refAccount",
            "record": {
              "attributes": {
                "type": "Account",
                "method": "GET",
                "id": "ExternalId__c/123"
              }
            }
          },
          {
            "referenceId": "refContact",
            "record": {
              "attributes": {
                "type": "Contact",
                "method": "GET",
                "id": "ExternalId__c/123"
              }
            }
          },
          {
            "referenceId": "refInvoice",
            "record": {
              "Attributes": {
                "type": "Invoice",
                "method": "POST"
              },
              "billingAccountId": "001SG00000mKFfGYAW",
              "billToContactId": "003SG00000Bnv5HYAR",
              "paymentTermId": "2OXxx0000004CFUGA2",
              "referenceEntityId": "801xx000003GeQQAA0",
              "status": "Draft",
              "invoiceDate": "2024-12-19",
              "currencyIsoCode": "USD",
              "dueDate": "2024-12-19",
              "invoiceNumber": "DOC-10",
              "description": "Sample Invoice",
              "uniqueIdentifier": "5873af8f-f007-4aa0-9e3d-53a08c3f59de"
            }
          },
          {
            "referenceId": "refBillingAddress",
```

```
      "record": {
        "Attributes": {
          "type": "InvoiceAddressGroup",
          "method": "POST"
        },
        "street": "123 Main St",
        "city": "NewYork",
        "postalCode": "10001",
        "state": "New York",
        "country": "US",
        "longitude": "123.456",
        "latitude": "78.910",
        "invoiceId": "3ttSG000000EPjJYAW"
      }
    },
    {
      "referenceId": "refShippingAddress",
      "record": {
        "attributes": {
          "type": "InvoiceAddressGroup",
          "method": "POST"
        },
        "street": "123 Main St",
        "city": "NewYork",
        "postalCode": "10001",
        "state": "New York",
        "country": "US",
        "longitude": "123.456",
        "latitude": "78.910",
        "invoiceId": "3ttSG000000EPjJYAW"
      }
    },
    {
      "referenceId": "refInvoiceLine1",
      "record": {
        "attributes": {
          "type": "InvoiceLine",
          "method": "POST"
        },
        "name": "productName1",
        "product2Id": "01txx0000006ic4AAA",
        "invoiceLineStartDate": "2024-11-10",
        "invoiceLineEndDate": "2024-11-13",
        "quantity": "10",
        "unitPrice": "10",
        "chargeAmount": "100",
        "invoiceId": "3ttSG000000EPjJYAW",
        "referenceEntityItemId": "802xx000001neB9AAI",
        "billingAddressId": "8zzSG0000000IbFYAU",
        "shippingAddressId": "8zzSG0000000IbGYAU",
        "taxTreatmentId": "1ttxx0000000BOTAA2",
        "legalEntityId": "0fwxx0000000001AAA",
        "legalEntityAccountingPeriodId": "1HLxx0000004C92GAE"
      }
```

```
    },
    {
      "referenceId": "refInvoiceLine2",
      "record": {
        "attributes": {
          "type": "InvoiceLine",
          "method": "POST"
        },
        "name": "productName2",
        "product2Id": "01txx0000006ic4AAA",
        "invoiceLineStartDate": "2024-11-10",
        "invoiceLineEndDate": "2024-11-15",
        "quantity": "10",
        "unitPrice": "10",
        "chargeAmount": "100",
        "invoiceId": "3ttSG000000EPjJYAW",
        "referenceEntityItemId": "802xx000001neB9AAI",
        "billingAddressId": "8zzSG0000000IbFYAU",
        "shippingAddressId": "8zzSG0000000IbGYAU",
        "taxTreatmentId": "1ttxx00000001DpAAI",
        "legalEntityId": "0fwxx0000000001AAA",
        "legalEntityAccountingPeriodId": "1HLxx0000004C92GAE"
      }
    },
    {
      "referenceId": "refInvoiceLineTax",
      "record": {
        "attributes": {
          "type": "InvoiceLineTax",
          "method": "POST"
        },
        "taxAmount": 7.25,
        "taxCode": "CA-94121",
        "taxName": "CALIFORNIA",
        "taxRate": 0.25,
        "taxEffectiveDate": "2024-11-10",
        "taxDocumentNumber": "123",
        "taxExemptAmount": 0,
        "taxTransactionNumber": "kl",
        "description": "Associated tax line.",
        "invoiceLineId": "5TVSG0000004dAr4AI"
      }
    },
    {
      "referenceId": "refInvoiceLineTax1",
      "record": {
        "attributes": {
          "type": "InvoiceLineTax",
          "method": "POST"
        },
        "taxAmount": "10",
        "taxCode": "CA-94121",
        "taxName": "CALIFORNIA",
        "taxRate": 0.25,
```

```
              "taxEffectiveDate": "2024-11-10",
              "taxDocumentNumber": "123",
              "taxExemptAmount": 0,
              "taxTransactionNumber": "125",
              "invoiceLineId": "5TVSG0000004xeD4AQ",
              "description": "Associated tax line."
            }
          }
        ]
      }
    }
  ]
}
```

This example shows a sample request ingest a draft invoice without a tax callout.

```
{
  "invoices": [
    {
      "taxCalculationStatus": "Estimated",
      "graph": {
        "graphId": "123",
        "records": [
          {
            "referenceId": "refAccount",
            "record": {
              "attributes": {
                "type": "Account",
                "method": "GET",
                "id": "TestExternalId__c/123"
              }
            }
          },
          {
            "referenceId": "refInvoice",
            "record": {
              "Attributes": {
                "type": "Invoice",
                "method": "POST"
              },
              "billingAccountId": "@{refAccount.Id}",
              "billToContactId": "003xx000004Wk8qAAC",
              "paymentTermId": "2OXxx0000004CFUGA2",
              "referenceEntityId": "801xx000003GeQQAA0",
              "status": "Draft",
              "invoiceDate": "2024-12-19",
              "currencyIsoCode": "USD",
              "dueDate": "2024-12-19",
              "invoiceNumber": "DOC-10",
              "description": "testInvoice",
              "uniqueIdentifier": "c76011a1-e113-49d9-9c54-3b5c68950ada"
            }
          },
          {
            "referenceId": "refBillingAddress",
```

```
      "record": {
        "Attributes": {
          "type": "InvoiceAddressGroup",
          "method": "POST"
        },
        "street": "123 Main St",
        "city": "NewYork",
        "postalCode": "10001",
        "state": "New York",
        "country": "US",
        "longitude": "123.456",
        "latitude": "78.910",
        "invoiceId": "@{refInvoice.id}"
      }
    },
    {
      "referenceId": "refShippingAddress",
      "record": {
        "attributes": {
          "type": "InvoiceAddressGroup",
          "method": "POST"
        },
        "street": "123 Main St",
        "city": "NewYork",
        "postalCode": "10001",
        "state": "New York",
        "country": "US",
        "longitude": "123.456",
        "latitude": "78.910",
        "invoiceId": "@{refInvoice.id}"
      }
    },
    {
      "referenceId": "refInvoiceLine1",
      "record": {
        "attributes": {
          "type": "InvoiceLine",
          "method": "POST"
        },
        "name": "productName1",
        "product2Id": "01txx0000006ic4AAA",
        "invoiceLineStartDate": "2024-11-10",
        "invoiceLineEndDate": "2024-11-13",
        "quantity": "10",
        "unitPrice": "10",
        "chargeAmount": "100",
        "invoiceId": "@{refInvoice.id}",
        "referenceEntityItemId": "802xx000001neB9AAI",
        "billingAddressId": "@{refBillingAddress.id}",
        "shippingAddressId": "@{refShippingAddress.id}",
        "taxTreatmentId": "1ttxx0000000BOTAA2",
        "legalEntityId": "0fwxx0000000001AAA",
        "legalEntityAccountingPeriodId": "1HLxx0000004C92GAE"
      }
```

```
    },
    {
      "referenceId": "refInvoiceLine2",
      "record": {
        "attributes": {
          "type": "InvoiceLine",
          "method": "POST"
        },
        "name": "productName2",
        "product2Id": "01txx0000006ic4AAA",
        "invoiceLineStartDate": "2024-11-10",
        "invoiceLineEndDate": "2024-11-15",
        "quantity": "10",
        "unitPrice": "10",
        "chargeAmount": "100",
        "invoiceId": "@{refInvoice.id}",
        "referenceEntityItemId": "802xx000001neB9AAI",
        "billingAddressId": "@{refBillingAddress.id}",
        "shippingAddressId": "@{refShippingAddress.id}",
        "taxTreatmentId": "1ttxx00000001DpAAI",
        "legalEntityId": "0fwxx0000000001AAA",
        "legalEntityAccountingPeriodId": "1HLxx0000004C92GAE"
      }
    },
    {
      "referenceId": "refInvoiceLineTax",
      "record": {
        "attributes": {
          "type": "InvoiceLineTax",
          "method": "POST"
        },
        "taxAmount": 7.25,
        "taxCode": "CA-94121",
        "taxName": "CALIFORNIA",
        "taxRate": 0.25,
        "taxEffectiveDate": "2024-11-10",
        "taxDocumentNumber": "123",
        "taxExemptAmount": 0,
        "taxTransactionNumber": "kl",
        "description": "description for tax Line",
        "invoiceLineId": "@{refInvoiceLine1.id}"
      }
    },
    {
      "referenceId": "refInvoiceLineTax1",
      "record": {
        "attributes": {
          "type": "InvoiceLineTax",
          "method": "POST"
        },
        "taxAmount": "10",
        "taxCode": "CA-94121",
        "taxName": "CALIFORNIA",
        "taxRate": 0.25,
```

```
              "taxEffectiveDate": "2024-11-10",
              "taxDocumentNumber": "123",
              "taxExemptAmount": 0,
              "taxTransactionNumber": "125",
              "invoiceLineId": "@{refInvoiceLine2.id}",
              "description": "description for tax Line"
            }
          }
        ]
      }
    }
  ]
}
```

This example shows a sample request to ingest posted invoices.

```
{
  "invoices": [
    {
      "taxCalculationStatus": "Posted",
      "graph": {
        "graphId": "123",
        "records": [
          {
            "referenceId": "refAccount",
            "record": {
              "attributes": {
                "type": "Account",
                "method": "GET",
                "id": "ExternalId__c/123"
              }
            }
          },
          {
            "referenceId": "refInvoice",
            "record": {
              "Attributes": {
                "type": "Invoice",
                "method": "POST"
              },
              "billingAccountId": "001SG00000njpF3YAI",
              "billToContactId": "003xx000004Wk8qAAC",
              "paymentTermId": "2OXxx0000004CFUGA2",
              "referenceEntityId": "801xx000003GeQQAA0",
              "status": "Posted",
              "invoiceDate": "2024-12-19",
              "currencyIsoCode": "USD",
              "dueDate": "2024-12-19",
              "invoiceNumber": "DOC-10",
              "description": "Sample Invoice",
              "uniqueIdentifier": "9994b2c4-c0c3-47c3-806f-ae6e1f16bac3"
            }
          },
          {
            "referenceId": "refBillingAddress",
```

```
      "record": {
        "Attributes": {
          "type": "InvoiceAddressGroup",
          "method": "POST"
        },
        "street": "123 Main St",
        "city": "NewYork",
        "postalCode": "10001",
        "state": "New York",
        "country": "US",
        "longitude": "123.456",
        "latitude": "78.910",
        "invoiceId": "3ttSG000000FFWbYAO"
      }
    },
    {
      "referenceId": "refShippingAddress",
      "record": {
        "attributes": {
          "type": "InvoiceAddressGroup",
          "method": "POST"
        },
        "street": "123 Main St",
        "city": "NewYork",
        "postalCode": "10001",
        "state": "New York",
        "country": "US",
        "longitude": "123.456",
        "latitude": "78.910",
        "invoiceId": "3ttSG000000FFWbYAO"
      }
    },
    {
      "referenceId": "refInvoiceLine1",
      "record": {
        "attributes": {
          "type": "InvoiceLine",
          "method": "POST"
        },
        "name": "productName1",
        "product2Id": "01txx0000006ic4AAA",
        "invoiceLineStartDate": "2024-11-10",
        "invoiceLineEndDate": "2024-11-13",
        "quantity": "10",
        "unitPrice": "10",
        "chargeAmount": "100",
        "invoiceId": "3ttSG000000FFWbYAO",
        "referenceEntityItemId": "802xx000001neB9AAI",
        "billingAddressId": "8zzSG0000000NmDYAU",
        "shippingAddressId": "8zzSG0000000IbGYAU",
        "taxTreatmentId": "1ttxx0000000BOTAA2",
        "legalEntityId": "0fwxx0000000001AAA",
        "legalEntityAccountingPeriodId": "1HLxx0000004C92GAE"
      }
```

```
          },
          {
            "referenceId": "refInvoiceLine2",
            "record": {
              "attributes": {
                "type": "InvoiceLine",
                "method": "POST"
              },
              "name": "productName2",
              "product2Id": "01txx0000006ic4AAA",
              "invoiceLineStartDate": "2024-11-10",
              "invoiceLineEndDate": "2024-11-15",
              "quantity": "10",
              "unitPrice": "10",
              "chargeAmount": "100",
              "invoiceId": "3ttSG000000FFWbYAO",
              "referenceEntityItemId": "802xx000001neB9AAI",
              "billingAddressId": "8zzSG0000000NmDYAU",
              "shippingAddressId": "8zzSG0000000IbGYAU",
              "taxTreatmentId": "1ttxx00000001DpAAI",
              "legalEntityId": "0fwxx0000000001AAA",
              "legalEntityAccountingPeriodId": "1HLxx0000004C92GAE"
            }
          },
          {
            "referenceId": "refInvoiceLineTax1",
            "record": {
              "attributes": {
                "type": "InvoiceLineTax",
                "method": "POST"
              },
              "taxAmount": "10",
              "taxCode": "CA-94121",
              "taxName": "CALIFORNIA",
              "taxRate": 0.25,
              "taxEffectiveDate": "2024-11-10",
              "taxDocumentNumber": "123",
              "taxExemptAmount": 0,
              "taxTransactionNumber": "125",
              "invoiceLineId": "5TVSG0000004mJ04AI",
              "description": "Associated tax line."
            }
          }
        ]
      }
    }
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`invoices`	Invoice Ingestion Input[]	List of invoices to be generated or ingested, enabling the submission and processing of multiple invoices in a single request. This API supports one invoice per request. To send 25 requests at a time, see the Composite Batch request.	Required	63.0

Response body for POST

Invoice Ingestion

Invoice Estimated Tax Calculation (POST)

Calculate estimated tax for invoices with invoice lines that have the `TaxProcessingStatus` as either `Pending` or `Estimated`.

Resource

```
/commerce/invoicing/invoices/collection/actions/calculate-estimated-tax
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/commerce/invoicing/invoices/collection/actions/calculate-estimated-tax
```

Available version

63.0

HTTP methods

POST

Request body for POST

JSON example

```
{
  "invoiceIds": ["3ttxx0000004CIjAAM"]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`correlationId`	String	Splunk correlation ID to track the messages that are related to the request and are logged in Splunk by the different services involved in the request. If the ID isn't specified, the service creates a random Universally Unique Identifier (UUID).	Optional	63.0

Name	Type	Description	Required or Optional	Available Version
`invoiceIds`	String[]	IDs of the invoices for which the estimated tax must be calculated. You can specify one invoice per API request.	Required	63.0

Response body for POST

Revenue Async Response

Invoice Preview (POST)

Generate preview invoices, which includes the estimated tax amounts, for a billing transaction for the next two billing periods.

Invoice preview provides details of the upcoming invoices for orders or quotes within the system.

Special Access Rules

You need the Billing Operations User or Billing Customer Service permission set to use this API. See Invoice Preview to learn more about the configuration prerequisites.

Resource

```
/commerce/invoicing/invoices/collection/actions/preview
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/commerce/invoicing/invoices/collection/actions/preview
```

Available version

63.0

HTTP methods

POST

Request body for POST

JSON example

```
{
    "billingTransactionId": "801Z600000004LoIAI",
    "previewDate": "2024-12-04"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`billingTransactionId`	String	ID of the record to generate the preview invoices for.	Required	63.0
`previewDate`	String	The date on which the preview invoice is generated. For the first invoice, the preview date is the target date for generating the invoice. For the second invoice, the target date is calculated	Optional	63.0

Name	Type	Description	Required or Optional	Available Version
		based on the preview date and the minimum billing frequency of the transactions. The default value is the current date.		

Response body for POST

Invoice Preview

Invoice Run Recovery (POST)

Recover records associated with a failed invoice run. Recovery is required only when billing schedules remain in the `Processing` or `Error` status.

Special Access Rules

To use this API, you need the Invoice Scheduler API permission set.

Resource

```
/commerce/invoicing/invoice-batch-runs/invoiceBatchRunId/actions/recover
```

The *`invoiceBatchRunId`* parameter is the ID of the failed invoice batch run record whose details you want to retrieve.

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/commerce/invoicing/invoice-batch-runs/5IRxx0000004TwGGAU/actions/recover
```

Available version

62.0

HTTP methods

POST

Response body for POST

Invoice Batch Run Recovery on page 1615

Negative Invoice Lines to Credit Conversion (POST)

Convert a list of invoice lines with a negative amount into a posted credit memo. This conversion is applicable for a single invoice at a time.

Keep these considerations in mind when you use this API.

- All invoice lines must be related to the same invoice.
- The invoice line must have a negative amount.
- The invoice line must not be a previously converted credit memo.
- The invoice must have the `Posted` status.
- The invoice must not have any active settlements such as credit applications.

Special Access Rules

You need the Credit Memo Operations User permission set to use this API.

Resource

```
/commerce/invoicing/invoices/invoiceId/actions/convert-to-credit
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/commerce/invoicing/invoices/3ttxx00000000XtAAI/actions/convert-to-credit
```

Available version

62.0

HTTP methods

POST

Path parameter for POST

Name	Type	Description	Required or Optional	Available Version
`invoiceId`	String	ID of the invoice whose negative invoice lines must be converted.	Required	62.0

Request body for POST

JSON example

```
{
"invoiceLines": ["5TVxx0000004C92GAE", "5TVxx0000004C93GAE"],
"description": "Convert negative invoice lines into credit",
"effectiveDate":"2022-05-18"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`description`	String	Description stamped on the credit memo that's created after the negative invoice line conversion.	Optional	62.0
`effectiveDate`	String	Date stamped on the credit memo that's created after the negative invoice line conversion.	Required	62.0
`invoiceLines`	String[]	List of the negative invoice lines along with the associated invoice line taxes.	Required	62.0

Response body for POST

Convert Negative Invoice Lines

Posted Invoice Voidance (POST)

Void a posted invoice to rebill the customer, if necessary.

This API request changes the invoice status from `Posted` to `Void In Progress`. The invoice remains in the `Void In Progress` status until the credit is applied and financial fields are recalculated on the invoice's related billing period items and billing schedule. The invoice status changes to `Voided` after all recalculations are completed.

Keep these considerations in mind when you use this API.

- The balance and total amount on the invoice must be equal. If these amounts aren't equal due to payments or credits, the API request fails.
- You can't call other APIs on an invoice with the `Void In Progress` status. You also can't update the invoice fields.
- You can void only the most recently posted invoice on a billing schedule.
- To void an invoice that has payments or credits, use the Credit Memo Unapply (POST) API.

Credit Memos

The void process creates a credit memo, which contains one credit memo line for each invoice line, including tax lines. For example, if the invoice line has a balance of US$20, the related credit memo line has a balance of $20. The credit memo's balance is then allocated to the invoice header's balance, reducing it to zero. A credit memo invoice application is created to record the details of the void process.

Negative Invoice Lines

If an invoice has negative invoice lines that aren't converted to a credit memo, you can use this endpoint to void the posted invoice.

Special Access Rules

You need the Void a Posted Invoice API permission set to use this API.

Resource

```
/commerce/invoicing/invoices/invoiceId/actions/void
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/commerce/invoicing/invoices/3ttxx00000000XtAAI/actions/void
```

Available version

62.0

HTTP methods

POST

Path parameter for POST

Properties

Name	Type	Description	Required or Optional	Available Version
`invoiceId`	String	ID of the posted invoice to be voided.	Required	62.0

Response body for POST

Revenue Async Response

Resume Billing (POST)

Resume billing for billing schedule groups or an account that's currently on hold.

Resource

```
/commerce/invoicing/actions/resume-billing
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/commerce/invoicing/actions/resume-billing
```

Available version

63.0

HTTP methods

POST

Request body for POST

JSON example

```
{
    "referenceIds":
    [
        {
          "referenceId": "001DU000001o2UwYAI",
          "resumeDate": "2024-11-27"
        }
    ]

}
```

Properties

Name	Type	Description	Required or Optional	Available Version
referenceIds	Resume Billing Object Input[]	Input representation of the account or billing schedule group IDs to resume the billing operation for.	Required	63.0

Response body for POST

Suspend Resume Billing

Suspend Billing (POST)

Suspend billing for billing schedule groups or an account for a predefined period.

Resource

```
/commerce/invoicing/actions/suspend-billing
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/commerce/invoicing/actions/suspend-billing
```

Available version

63.0

HTTP methods

POST

Request body for POST

JSON example

```
{
    "referenceIds":
    [
        {
            "referenceId": "001DU000001o2UwYAI",
            "suspendDate": "2024-11-27",
            "resumeDate": "2024-12-27"
        }
    ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`referenceIds`	Suspend Billing Object Input[]	Input representation of the account or billing schedule group IDs to suspend the billing operation for.	Required	63.0

Response body for POST

Suspend Resume Billing

Standalone Credit Memo (POST)

Create a credit memo without applying it to an invoice. You can credit the invoice at a later date.

In the API request, specify the credit memo header information, charge parameters, adjustment parameters, and tax parameters. A credit memo requires at least one credit memo line. The credit memo line can be a charge or an adjustment.

Specify the credit memo lines that you want as lists of charges and adjustments. Each credit memo line must be related to a product.

Special Access Rules

You need the Credit Memo Operations User permission set to use this API.

Resource

```
/commerce/invoicing/credit-memos/actions/generate
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/commerce/invoicing/credit-memos/actions/generate
```

Available version

62.0

HTTP methods

POST

Request body for POST

JSON example

This example shows a sample request with the `Ignore` tax strategy.

```
{
  "billingAccountId": "001j000000WCFB800x",
  "type": "Posted",
  "description": "Standalone credit memo with ignored tax.",
  "currencyIsoCode": "USD",
  "taxStrategy": "Ignore",
  "charges": [
    {
      "chargeAmount": 100,
      "productId": "01tR0000000njDiIAI"
    }
  ]
}
```

This example shows a sample request with the `Calculate` tax strategy for a charge line.

```
{
  "billingAccountId": "001DU000001nhoPYAQ",
  "description": "Standalone Credit Memo",
  "type": "Posted",
  "currencyIsoCode": "USD",
  "taxStrategy": "Ignore",
  "charges": [
    {
      "productId": "01tR0000000njDiIAI",
      "chargeAmount": 10,
      "taxStrategy": "Calculate",
      "treatmentId": "1ttxx00000001VZAAY"
    }
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`billTo ContactId`	String	Contact related to the credit memo.	Optional	62.0
`billing AccountId`	String	ID of the account that the credit is issued to.	Required	62.0
`charges`	Standalone Credit Memo Charge Input[]	Charge lines of the credit memo. Requires at least one charge line.	Required	62.0
`currency IsoCode`	String	ISO code currency of the new credit that's issued.	Optional	62.0

Name	Type	Description	Required or Optional	Available Version
`description`	String	Description for the new credit that's issued.	Optional	62.0
`effectiveDate`	String	Effective date of the credit memo. If the value isn't specified, then it's null.	Optional	62.0
`externalReference`	String	ID of the external reference for the credit memo.	Optional	62.0
~~`externalReference`~~ `DataSource`	String	Source of the external reference for the credit memo.	Optional	62.0
`taxEffectiveDate`	String	Effective date of the credit memo tax. If the value isn't specified, then it's null.	Optional	62.0
`taxStrategy`	String	Specifies how tax lines must be created for the standalone credit memos. Valid values are: • `Ignore`—Specifies that the creation of tax lines must be ignored. • `Calculate`—Specifies that tax must be calculated by using the API.	Required	62.0
`type`	String	Type of credit memo to be created. Valid value is `Posted`.	Optional	62.0

Response body for POST

Revenue Async Response

Tax Calculation (POST)

Calculate tax for a transaction.

Special Access Rules

To use this API, you need the CalculateTaxes API permission set.

Resource

```
/commerce/taxes/actions/calculate
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v63.0/commerce/taxes/actions/calculate
```

Available version

62.0

HTTP methods

POST

Request body for POST

JSON example

```
{
  "addresses": {
    "billTo": {
      "street": "123 Main Street",
      "city": "Bainbridge Island",
      "state": "WA",
      "postalCode": "98110",
      "country": "US"
    },
    "soldTo": {
      "street": "123 Main Street",
      "city": "Bainbridge Island",
      "state": "WA",
      "postalCode": "98110",
      "country": "US"
    },
    "shipFrom": {
      "street": "123 Alaskan Way",
      "city": "Seattle",
      "state": "WA",
      "country": "US",
      "postalCode": "98101"
    },
    "shipTo": {
      "street": "123 Main street",
      "city": "Bainbridge Island",
      "state": "WA",
      "postalCode": "98110",
      "country": "US"
    }
  },
  "currencyIsoCode": "USD",
  "customerDetails": {
    "accountId": "001xx000003HYD5AAO"
  },
  "description": "Monthly invoice for account 001xx000003HYEhAAO",
  "documentCode": "INV-003",
  "effectiveDate": "2022-03-09T10:30:41.000Z",
  "isCommit": true,
  "lineItems": [
    {
      "quantity": 1,
      "amount": 100,
      "taxCode": "TX0001",
      "productCode": "Y0001",
      "description": "New Product",
      "effectiveDate": "2022-03-09T10:30:41.000Z",
      "lineNumber": "5TVxx0000004C92GAE",
      "addresses": {
        "shipFrom": {
          "street": "123 Alaskan Way",
```

```
          "city": "Seattle",
          "state": "WA",
          "country": "US",
          "postalCode": "98101",
          "latitude": 45.12,
          "longitude": 45.12
        },
        "shipTo": {
          "street": "123 Auburn Blvd",
          "city": "Sacramento",
          "state": "CA",
          "country": "US",
          "postalCode": "95841",
          "latitude": 45.12,
          "longitude": 45.12
        },
        "soldTo": {
          "street": "123 Auburn Blvd",
          "city": "Sacramento",
          "state": "CA",
          "country": "US",
          "postalCode": "95841",
          "latitude": 45.12,
          "longitude": 45.12
        },
        "billTo": {
          "street": "123 Auburn Blvd",
          "city": "Sacramento",
          "state": "CA",
          "country": "US",
          "postalCode": "95841",
          "latitude": 45.12,
          "longitude": 45.12
        }
      }
    }
  ],
  "referenceEntityId": "001xx000003HYEhAAO",
  "taxEngineId": "4wAxx0000000g8v",
  "taxType": "Actual",
  "taxTransactionType": "Debit",
  "transactionDate": "2022-03-09T10:30:41.000Z"
}
```


Properties

Name	Type	Description	Required or Optional	Available Version
addresses	Addresses	Address details for tax calculation.	Optional	62.0
currencyIsoCode	String	Currency ISO code that's used for tax calculation.	Optional	62.0

Name	Type	Description	Required or Optional	Available Version
customerDetails	Customer Details	Customer details for determining the applicable tax.	Optional	62.0
description	String	Description of the tax transaction.	Optional	62.0
documentCode	String	Unique identifier for the tax document. If the documentCode property isn't specified, the tax engine auto-generates it.	Optional	62.0
effectiveDate	String	Date when the tax rules are applied. If a tax rate changes on a specific date, the effectiveDate property ensures the correct rate is applied based on the transaction's timing.	Optional	62.0
isCommit	Boolean	Indicates whether to submit the transaction to the tax engine for reporting (true) or not (false). This property value is true for invoices and credit memos with the Posted status.	Optional	62.0
lineItems	Line Item Input []	Details of the line items for calculating the applicable tax.	Required	62.0
reference DocumentCode	String	Reference document code. For subsequent transactions such as a credit tax, this property value specifies the original document code.	Optional	62.0
reference EntityId	String	ID of the related quote, invoice, and other transaction documents.	Optional	62.0
sellerDetails	Seller Details Input	Seller details for tax calculation.	Optional	62.0
taxEngineId	String	ID of the tax engine that's used to calculate tax.	Required	62.0
tax TransactionType	String	Type of the tax transaction. Valid values are: • Debit—Increases tax liability for the seller, requiring the seller to pay tax on the transaction. • Credit—Decreases tax liability for the seller, resulting in a tax refund for the seller. The default value is Debit.	Optional	62.0

Name	Type	Description	Required or Optional	Available Version
`taxType`	String	Type of the tax. Valid values are: • `Actual`—Exact tax amount that's calculated based on actual sales. • `Estimated`— Estimated tax amount, which is adjusted later to match actual tax calculations. For draft invoices and quote records, the tax type is marked as `Estimated`. After draft invoices are posted and the status changes to `Posted`, the tax type is updated to `Actual`. Similarly, when a quote record is finalized and converted into an actual order, the tax type is also updated to `Actual`.	Required	62.0
`transactionDate`	String	Date of the transaction that appears on the invoice, order, and other transaction documents.	Required	62.0

Response body for POST

Tax Calculation

Request Bodies

Learn more about the available request bodies of Billing APIs.

Address Input

Input representation of the details of an address.

Addresses Input

Input representation of the details of the addresses for calculating tax.

Batch Invoice Filter Criteria Input

Input representation of the filter criteria for an invoice batch run.

Batch Invoice Scheduler Input

Input representation of the details of the request to create an invoice scheduler.

Billing Schedule Recovery Input

Input representation of the details of the billing schedules to recover the associated invoice.

Context-Aware Billing Schedule

Input representation of the billing transaction details.

Convert Negative Invoice Lines Input

Input representation of the details of the request to convert a list of negative invoice lines into a credit.

Credit Invoice Input

Input representation of the details of the request to create a credit memo.

Credit Invoice Line Input

Input representation of the details of the invoice lines to be credited.

Credit Invoice Line Tax Input

Input representation of the details of the tax lines to be created manually for the invoice line.

Credit Memo Addresses Input

Input representation of the details of the billing and shipping addresses.

Credit Memo Apply Input

Input representation of the request to apply a credit memo to an invoice.

Credit Memo Apply Application Input

Input representation of the request to specify one or more applications to apply a credit memo for, with each application representing an invoice.

Credit Memo Unapply Input

Input representation of the request to unapply a credit memo from an invoice.

Credit Memo Line Apply Input

Input representation of the details of the request to apply a credit memo line to an invoice line.

Credit Memo Line Application Input

Input representation of the request to specify one or more applications to apply a credit memo line for, with each application representing an invoice line.

Credit Memo Line Unapply Input

Input representation of the details of the request to unapply a credit memo line from an invoice line.

Customer Details Input

Input representation of the customer details for tax calculation.

Frequency Cadence Options

Input representation of the frequency cadence options for an invoice scheduler.

Graph Record for Invoice Ingestion

A Graph record is an object that's a part of the graph structure, representing both the fields and relationships among different objects. Each record in the graph can contain attributes, which are fields of the object, and references to other related records.

Invoice Draft To Posted Input

Input representation of the details of the draft invoice that's posted.

Invoice Estimated Tax Calculation Input

Details of the invoice for which the estimated tadsfsefx must be calculated.

Invoice Ingestion Input

Input representation of the details of the invoice to be processed. The details include the tax processing status, user preferences for tax callouts, and associated object graph representation.

Invoice Input for Ingestion

Input representation of the details of the invoice that must be generated for or ingested into Billing.

Invoice Input

Input representation of the details of the billing schedule.

Invoice Preview Input

Input representation of the details of the billing transaction that the preview invoices are generated for.

Line Item Input

Input representation of the details of the line item for tax calculation.

Resume Billing Input

Input representation of the details of the request to resume the billing operation for an account or a billing schedule group.

Resume Billing Object Input

Input representation of the details such as the ID of the account or billing schedule group along with the effective date. These details are used to start the billing operation.

Seller Details Input

Input representation of the seller details for tax calculation.

Standalone Credit Memo Charge Input

Input representation of the details of the charge lines of a credit memo.

Standalone Credit Memo Input

Input representation of the details required to create a standalone credit memo.

Standalone Credit Memo Tax Input

Input representation of the details of the tax request.

Suspend Billing Input

Input representation of the details of the request to suspend the billing operation for an account or a billing schedule group.

Suspend Billing Object Input

Input representation of the details such as the ID of the account or billing schedule group along with the effective dates. These details are used to suspend the billing operation.

Tax Calculation Input

Input representation of the details of the request to calculate tax.

Void Posted Invoice Input

Input representation of the details of the invoice to be voided.

Address Input

Input representation of the details of an address.

JSON example

JSON example

```
"billingAddress": {
  "street": "1 Market St #300",
  "city": "San Francisco",
  "state": "CA",
  "country": "US",
  "postalCode": "94105",
  "latitude": "37.789901",
  "longitude": "-122.396923"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`city`	String	Address city.	Optional	62.0
`country`	String	Address country.	Optional	62.0
`latitude`	Double	Latitude for the address.	Optional	62.0
`locationCode`	String	Location code for the address.	Optional	62.0
`longitude`	Double	Longitude for the address.	Optional	62.0
`postalCode`	String	Postal code for the address.	Optional	62.0
`state`	String	Address state.	Optional	62.0
`street`	String	Address street.	Optional	62.0

Addresses Input

Input representation of the details of the addresses for calculating tax.

JSON example

```
{
  "addresses": {
    "billTo": {
      "street": "123 Main Street",
      "city": "Bainbridge Island",
      "state": "WA",
      "postalCode": "98110",
      "country": "US"
    },
    "soldTo": {
      "street": "123 Main Street",
      "city": "Bainbridge Island",
      "state": "WA",
      "postalCode": "98110",
      "country": "US"
    },
    "shipFrom": {
      "street": "123 Alaskan Way",
      "city": "Seattle",
      "state": "WA",
      "country": "US",
      "postalCode": "98101"
    },
    "shipTo": {
      "street": "123 Main street",
      "city": "Bainbridge Island",
      "state": "WA",
      "postalCode": "98110",
```

```
      "country": "US"
    }
  }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
billTo	Address Input	Billing address of the item.	Optional	62.0
shipFrom	Address Input	Address that the item is shipped from.	Optional	62.0
shipTo	Address Input	Address that the item is shipped to.	Required	62.0
soldTo	Address Input	Address of the entity that the item is sold to.	Optional	62.0

Batch Invoice Filter Criteria Input

Input representation of the filter criteria for an invoice batch run.

JSON example

```
    "filterCriteria": [
        {
            "operation": "InList",
            "value": "Batch 2,Batch 3,Batch 4",
            "criteriaSequence": 1,
            "objectName": "BillingSchedule",
            "fieldName": "InvoiceRunMatchingValue"
        },
        {
            "operation": "Equals",
            "value": "001xx000003GZG5AAO",
            "criteriaSequence": 2,
            "objectName": "BillingSchedule",
            "fieldName": "BillingAccount"
        },
        {
            "operation": "Equals",
            "value": "0fwxx0000000001AAA",
            "criteriaSequence": 3,
            "objectName": "BillingScheduleGroup",
            "fieldName": "LegalEntity"
        },
        {
            "operation": "InList",
            "value": "OneTime,Recurring",
            "criteriaSequence": 4,
            "objectName": "BillingSchedule",
            "fieldName": "BillingTermUnit"
        },
        {
```

```
            "operation": "Equals",
            "value": "USD",
            "criteriaSequence": 5,
            "objectName": "BillingSchedule",
            "fieldName": "Currency_Iso_code"
        }
    ]
```

Properties

Name	Type	Description	Required or Optional	Available Version
`fieldName`	String	Name of the field that this filter is applicable for. Valid values are: • `Currency_Iso_code`—If multiple currencies are enabled for the org, then this value is required. • `InvoiceRunMatchingValue` • `BillingAccount` • `LegalEntity` • `BillingTermUnit`	Required	62.0
`objectName`	Object	Name of the object that the filter is applicable for.	Required	62.0
`operation`	String	Operation to be performed for comparison. Valid values are: • `Equals` • `InList`—This value supports only the `InvoiceRunMatchingValue` and `BillingTermUnit` fields with API version 62.0 and later. In addition, this value supports the `CurrencyIsoCode` field with API version 63.0 and later.	Required	62.0
`value`	String	Value for the filter criteria.	Required	62.0

Batch Invoice Scheduler Input

Input representation of the details of the request to create an invoice scheduler.

JSON example

This example shows a sample request to create an invoice scheduler that generates invoices once.

```
{
  "schedulerName": "InvoiceScheduler",
  "startDate": "2024-05-06",
```

```
  "invoiceStatus": "POSTED",
  "preferredTime": "00:45",
  "targetDate": "2024-05-22",
  "invoiceDate": "2024-05-22",
  "frequencyCadence": "Once",
  "frequencyCadenceOptions": {},
  "timezone": "Asia/Kolkata",
  "status": "Active",
  "filterCriteria": [
    {
      "operation": "InList",
      "value": "Batch 2,Batch 3,Batch 4",
      "criteriaSequence": 1,
      "objectName": "BillingSchedule",
      "fieldName": "InvoiceRunMatchingValue"
    },
    {
      "operation": "Equals",
      "value": "001xx000003GZG5AAO",
      "criteriaSequence": 2,
      "objectName": "BillingSchedule",
      "fieldName": "BillingAccount"
    },
    {
      "operation": "Equals",
      "value": "0fwxx0000000001AAA",
      "criteriaSequence": 3,
      "objectName": "BillingScheduleGroup",
      "fieldName": "LegalEntity"
    },
    {
      "operation": "InList",
      "value": "OneTime,Recurring",
      "criteriaSequence": 4,
      "objectName": "BillingSchedule",
      "fieldName": "BillingTermUnit"
    },
    {
      "operation": "Equals",
      "value": "USD",
      "criteriaSequence": 5,
      "objectName": "BillingSchedule",
      "fieldName": "Currency_Iso_code"
    }
  ]
}
```

This example shows a sample request to create an invoice scheduler that generates invoices daily.

```
{
  "schedulerName": "InvoiceScheduler",
  "startDate": "2024-05-06",
  "endDate": "2026-05-06",
  "invoiceStatus": "POSTED",
  "preferredTime": "00:45",
```

```
  "targetDateOffset": 0,
  "invoiceDateOffset": 0,
  "isInvoiceDateFromRunDate": true,
  "frequencyCadence": "Daily",
  "frequencyCadenceOptions": {},
  "timezone": "Asia/Kolkata",
  "status": "Active",
  "filterCriteria": [
    {
      "operation": "InList",
      "value": "Batch 2,Batch 3,Batch 4",
      "criteriaSequence": 1,
      "objectName": "BillingSchedule",
      "fieldName": "InvoiceRunMatchingValue"
    },
    {
      "operation": "Equals",
      "value": "001xx000003GZG5AAO",
      "criteriaSequence": 2,
      "objectName": "BillingSchedule",
      "fieldName": "BillingAccount"
    },
    {
      "operation": "Equals",
      "value": "0fwxx0000000001AAA",
      "criteriaSequence": 3,
      "objectName": "BillingScheduleGroup",
      "fieldName": "LegalEntity"
    },
    {
      "operation": "InList",
      "value": "OneTime,Recurring",
      "criteriaSequence": 4,
      "objectName": "BillingSchedule",
      "fieldName": "BillingTermUnit"
    },
    {
      "operation": "InList",
      "value": "USD",
      "criteriaSequence": 5,
      "objectName": "BillingSchedule",
      "fieldName": "Currency_Iso_code"
    }
  ]
}
```

This example shows a sample request to create an invoice scheduler that generates invoices weekly.

```
{
  "schedulerName": "InvoiceScheduler",
  "startDate": "2024-05-06",
  "endDate": "2026-05-06",
  "invoiceStatus": "POSTED",
  "preferredTime": "00:45",
  "targetDateOffset": 0,
```

```
  "invoiceDateOffset": 0,
  "isInvoiceDateFromRunDate": false,
  "frequencyCadence": "Weekly",
  "frequencyCadenceOptions": {
    "recursOnDay": "Sunday"
  },
  "timezone": "Asia/Kolkata",
  "status": "Active",
  "filterCriteria": [
    {
      "operation": "InList",
      "value": "Batch 2,Batch 3,Batch 4",
      "criteriaSequence": 1,
      "objectName": "BillingSchedule",
      "fieldName": "InvoiceRunMatchingValue"
    },
    {
      "operation": "Equals",
      "value": "001xx000003GZG5AAO",
      "criteriaSequence": 2,
      "objectName": "BillingSchedule",
      "fieldName": "BillingAccount"
    },
    {
      "operation": "Equals",
      "value": "0fwxx0000000001AAA",
      "criteriaSequence": 3,
      "objectName": "BillingScheduleGroup",
      "fieldName": "LegalEntity"
    },
    {
      "operation": "InList",
      "value": "OneTime,Recurring",
      "criteriaSequence": 4,
      "objectName": "BillingSchedule",
      "fieldName": "BillingTermUnit"
    },
    {
      "operation": "Equals",
      "value": "USD",
      "criteriaSequence": 5,
      "objectName": "BillingSchedule",
      "fieldName": "Currency_Iso_code"
    }
  ]
}
```

This example shows a sample request to create an invoice scheduler that generates invoices monthly on a specific date.

```
{
  "schedulerName": "InvoiceScheduler",
  "startDate": "2024-05-06",
  "endDate": "2026-05-06",
  "invoiceStatus": "POSTED",
  "preferredTime": "00:45",
```

```
  "targetDateOffset": 0,
  "invoiceDateOffset": 0,
  "isInvoiceDateFromRunDate": false,
  "frequencyCadence": "Monthly",
  "frequencyCadenceOptions": {
    "recurringSubType": "SpecificDate",
    "recursOnDate": "L-1",
    "shouldExcludeWkendAndHldy": true
  },
  "timezone": "Asia/Kolkata",
  "status": "Active",
  "filterCriteria": [
    {
      "operation": "InList",
      "value": "Batch 2,Batch 3,Batch 4",
      "criteriaSequence": 1,
      "objectName": "BillingSchedule",
      "fieldName": "InvoiceRunMatchingValue"
    },
    {
      "operation": "Equals",
      "value": "001xx000003GZG5AAO",
      "criteriaSequence": 2,
      "objectName": "BillingSchedule",
      "fieldName": "BillingAccount"
    },
    {
      "operation": "Equals",
      "value": "0fwxx0000000001AAA",
      "criteriaSequence": 3,
      "objectName": "BillingScheduleGroup",
      "fieldName": "LegalEntity"
    },
    {
      "operation": "InList",
      "value": "OneTime,Recurring",
      "criteriaSequence": 4,
      "objectName": "BillingSchedule",
      "fieldName": "BillingTermUnit"
    },
    {
      "operation": "Equals",
      "value": "USD",
      "criteriaSequence": 5,
      "objectName": "BillingSchedule",
      "fieldName": "Currency_Iso_code"
    }
  ]
}
```

This example shows a sample request to create an invoice scheduler that runs immediately.

```
{
    "schedulerName": "InvoiceScheduler",
    "status": "Draft",
```

```
    "invoiceStatus": "Posted",
    "frequencyCadenceOptions": {
        "shouldStartRunImmediately": true
    },
    "frequencyCadence": "Once",
    "targetDate": "2024-08-28",
    "invoiceDate": "2024-08-28",
    "filterCriteria": [
        {
            "operation": "InList",
            "value": "Batch 2,Batch 3,Batch 4",
            "criteriaSequence": 1,
            "objectName": "BillingSchedule",
            "fieldName": "InvoiceRunMatchingValue"
        },
        {
            "operation": "Equals",
            "value": "001xx000003GZG5AAO",
            "criteriaSequence": 2,
            "objectName": "BillingSchedule",
            "fieldName": "BillingAccount"
        },
        {
            "operation": "Equals",
            "value": "0fwxx0000000001AAA",
            "criteriaSequence": 3,
            "objectName": "BillingScheduleGroup",
            "fieldName": "LegalEntity"
        },
        {
            "operation": "InList",
            "value": "OneTime,Recurring",
            "criteriaSequence": 4,
            "objectName": "BillingSchedule",
            "fieldName": "BillingTermUnit"
        },
        {
            "operation": "Equals",
            "value": "USD",
            "criteriaSequence": 5,
            "objectName": "BillingSchedule",
            "fieldName": "Currency_Iso_code"
        }
    ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
endDate	String	End date of the invoice scheduler.	Optional	63.0
filterCriteria	Batch Invoice Filter Criteria Input[]	List of line items of the filter criteria.	Optional	62.0

Name	Type	Description	Required or Optional	Available Version
`frequencyCadence`	String	Frequency to run the invoice scheduler. Valid values are: • `Once` • `Daily` • `Weekly` • `Monthly`	Required	62.0
`frequencyCadenceOptions`	Frequency Cadence Options	Frequency cadence options for the invoice scheduler.	Required	62.0
`invoiceDate`	String	Date shown on the invoice. This date is also used for tax calculations.	Required if the `frequencyCadence` property is set to `Once`.	62.0
`invoiceDateOffset`	Integer	Offset applied to the target date, which is the number of days added to or subtracted from the invoice date, to calculate the updated invoice date.	Required if the `frequencyCadence` property is set to `Daily`, `Weekly`, or `Monthly`.	62.0
`invoiceStatus`	String	Status of the invoice that specifies the expected invoice status from an invoice batch run. Valid values are: • `Draft` • `Posted`	Required	62.0
`isInvoiceDateFromRunDate`	Boolean	Indicates whether the invoice date is applicable from the date when the invoice scheduler is run (`true`) or not (`false`).	Optional	63.0
`preferredTime`	String	Preferred time for the invoice batch run.	Required	62.0
`schedulerName`	String	Name of the invoice scheduler, which must be unique in your org.	Required	62.0
`startDate`	String	Start date of the invoice scheduler.	Required	62.0
`status`	String	Status of the invoice scheduler. Valid values are: • `Draft` • `Active` • `Inactive`	Required	62.0

Name	Type	Description	Required or Optional	Available Version
`targetDate`	String	Target date of the invoice batch run. Billing schedules having the next billing date before this date are picked up for invoicing. The target date must be less than or equal to the maximum allowed target date for the org.	Required if the `frequencyCadence` property is set to `Once`.	62.0
`targetDateOffset`	Integer	Target date offset applied to the next invoice run date to calculate the target date. The offset is the number of days added to or subtracted from the next billing date.	Required if the `frequencyCadence` property is set to `Daily`, `Weekly`, or `Monthly`.	62.0
`timezone`	String	Time zone that's applicable for the invoice scheduler.	Optional	62.0

Billing Schedule Recovery Input

Input representation of the details of the billing schedules to recover the associated invoice.

JSON example

```
{
  "billingScheduleIds": ["44bDU00000000XXYAY"]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`billingScheduleIds`	String[]	IDs of the billing schedules to recover the invoice for. You can recover one billing schedule per API request.	Required	62.0

Context-Aware Billing Schedule

Input representation of the billing transaction details.

JSON example

```
{
    "billingTransactionIds": [ "801xx000003H1H9AAK"]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`billingTransactionIds`	String[]	ID of the billing transaction. This property value is the ID of the order if the source of the billing request is for the Order object. If the order product associated with the specified order ID doesn't have an associated billing treatment ID, the API considers the default billing treatment ID. The generated billing schedule group has the default billing treatment ID. The API supports only one billing transaction ID in the input.	Required	62.0

Convert Negative Invoice Lines Input

Input representation of the details of the request to convert a list of negative invoice lines into a credit.

JSON example

```
{
"invoiceLines": ["5TVxx0000004C92GAE", "5TVxx0000004C93GAE"],
"description": "Convert negative invoice lines into credit",
"effectiveDate":"2022-05-18"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`description`	String	Description stamped on the credit memo that's created after the negative invoice line conversion.	Optional	62.0
`effectiveDate`	String	Date stamped on the credit memo that's created after the negative invoice line conversion.	Required	62.0
`invoiceLines`	String[]	List of the negative invoice lines along with the associated invoice line taxes.	Required	62.0

Credit Invoice Input

Input representation of the details of the request to create a credit memo.

JSON example

This example shows a sample request with the `Calculate` tax strategy.

```
{
  "type": "POSTED",
  "taxStrategy": "Calculate",
  "taxEffectiveDate": "2024-07-20",
  "effectiveDate": "2024-07-20",
  "description": "Credit Invoice",
  "invoiceLines": [
    {
      "invoiceLineId": "5TVR00000004SiqOBE",
      "amountToCredit": 5
    }
  ]
}
```

This example shows a sample request with the `CopyFromInvoiceLine` tax strategy.

```
{
  "type": "POSTED",
  "taxStrategy": "CopyFromInvoiceLine",
  "effectiveDate": "2020-05-22",
  "description": "Credit Invoice",
  "invoiceLines": [
    {
      "invoiceLineId": "5TVR00000004SiqOBE",
      "amountToCredit": "5",
      "taxStrategy": "CopyFromInvoiceLine"
    }
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`description`	String	Description for the credit memo to be created.	Optional	62.0
`effectiveDate`	String	Date when the credit memo takes effect.	Optional	62.0
`invoiceLines`	Credit Invoice Line Input[]	List of the invoice lines to be credited. The invoice line IDs must be related to the invoice ID specified in the API request. If invoice lines aren't specified, the API request results in an error.	Required	62.0
`taxEffectiveDate`	String	Date when the tax takes effect to recalculate the taxes.	Optional	62.0
`taxStrategy`	String	Tax strategy to be applied across invoice lines. You can override the tax strategy at	Required	62.0

Name	Type	Description	Required or Optional	Available Version
		the individual invoice line level or at the tax line level. Valid values are: • `Ignore`—Specifies that the creation of tax lines must be ignored. • `CopyFromInvoiceLine`—Specifies that tax values must be copied from the invoice line. • `Calculate`—Specifies that tax must be calculated by using the API.		
`type`	String	Type of credit memo to be created. Valid value is `Posted`.	Optional	62.0

Credit Invoice Line Input

Input representation of the details of the invoice lines to be credited.

JSON example

```
"invoiceLines": [
  {
    "invoiceLineId": "5TVR00000004SiqOBE",
    "amountToCredit": 100,
    "taxStrategy": "ManualOverride",
    "taxEffectiveDate": "2021-08-01T21:22:41.000Z",
    "taxes": [
      {
        "taxAmount": 15,
        "taxName": "abc",
        "taxCode": "taxCode",
        "taxRate": 7
      }
    ],
    "addresses": {
      "billingAddress": {
        "street": "1 Market St #300",
        "city": "San Francisco",
        "state": "CA",
        "country": "US",
        "postalCode": "94105",
        "latitude": "37.789901",
        "longitude": "-122.396923"
      },
      "shippingAddress": {
        "street": "415 Mission St",
        "city": "San Francisco",
        "state": "CA",
        "country": "US",
        "postalCode": "94105",
```

```
          "latitude": "37.789901",
          "longitude": "-122.396923"
        }
      }
    },
    {
      "invoiceLineId": "5TVR00000004SiqOAE",
      "amountToCredit": 200,
      "taxStrategy": "CopyFromInvoiceLine"
    }
  ]
```

Properties

Name	Type	Description	Required or Optional	Available Version
`addresses`	Credit Memo Addresses Input[]	Addresses to be created manually for this invoice line and the overridden tax lines. These addresses are only applicable if this invoice line is using the `ManualOverride` tax strategy.	Optional	62.0
`amountToCredit`	Double	Amount to be credited from this invoice line.	Required	62.0
`invoiceLineId`	String	ID of the invoice line record to be credited. The invoice line ID must be related to the invoice ID specified in the API request.	Required	62.0
`isTaxOnlyCredit`	Boolean	Indicates whether the applicable tax amount is credited for the charge or adjustment amount (`true`), or the applicable tax amount is credited along with the charge or adjustment amount (`false`). The default value is `false`.	Optional	62.0
`taxEffectiveDate`	String	Date when the tax takes effect and the invoice line is credited.	Optional	62.0
`taxStrategy`	String	Tax strategy for crediting the invoice line. This tax strategy takes precedence over the `taxStrategy` property value specified in the Credit Invoice Input. Valid values are: • `Ignore`—Specifies that the creation of tax lines must be ignored. • `CopyFromInvoiceLine`—Specifies that tax values must be copied from the invoice line. • `Calculate`—Specifies that tax must be calculated by using the API.	Optional	62.0

Name	Type	Description	Required or Optional	Available Version
`taxes`	Credit Invoice Line Tax Input[]	List of tax lines to be created manually for this invoice line.	Required if the `taxStrategy` property value is `ManualOverride`.	62.0

Credit Invoice Line Tax Input

Input representation of the details of the tax lines to be created manually for the invoice line.

JSON example

```
      "taxes": [
        {
          "taxAmount": 15,
          "taxName": "abc",
          "taxCode": "taxCode",
          "taxRate": 7
        }
      ]
```

Properties

Name	Type	Description	Required or Optional	Available Version
`taxAmount`	Double	Amount of tax to be applied related to this invoice line.	Required	62.0
`taxCode`	String	Tax code to be applied related to this invoice line to create the tax line.	Optional	62.0
`taxName`	String	Name of tax to be applied related to this invoice line.	Optional	62.0
`taxRate`	Double	Tax rate used to create the tax line.	Optional	62.0

Credit Memo Addresses Input

Input representation of the details of the billing and shipping addresses.

JSON example

```
{
  "billingAddress": {
    "street": "1 Market St #300",
    "city": "San Francisco",
    "state": "CA",
    "country": "US",
    "postalCode": "94105",
    "latitude": "37.789901",
    "longitude": "-122.396923"
```

```
  },
  "shippingAddress": {
    "street": "415 Mission St",
    "city": "San Francisco",
    "state": "CA",
    "country": "US",
    "postalCode": "94105",
    "latitude": "37.789901",
    "longitude": "-122.396923"
  }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
billingAddress	Address Input[]	Billing address for charge or adjustment line.	Optional	62.0
shippingAddress	Address Input[]	Shipping address for charge or adjustment line.	Optional	62.0

Credit Memo Apply Input

Input representation of the request to apply a credit memo to an invoice.

JSON example

```
{
  "applications": [
    {
      "appliedToId": "3ttxx000000003FAAQ",
      "amount": 10,
      "description": "Apply to invoice for refund",
      "effectiveDate": "2024-07-01"
    },
    {
      "appliedToId": "3ttxx0000000001AAA",
      "amount": 100
    }
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
applications	Credit Memo Apply Application Input[]	List of one or more applications to apply the credit memo for. Each application represents an invoice that's credited by using the balance of the specified credit memo.	Required	62.0

Credit Memo Apply Application Input

Input representation of the request to specify one or more applications to apply a credit memo for, with each application representing an invoice.

JSON example

```
"applications": [
  {
    "appliedToId": "3ttxx000000003FAAQ",
    "amount": 10,
    "description": "Apply to invoice for refund",
    "effectiveDate": "2024-07-01"
  },
  {
    "appliedToId": "3ttxx0000000001AAA",
    "amount": 100
  }
]
```

Properties

Name	Type	Description	Required or Optional	Available Version
amount	Double	Credit amount to be applied to the invoice.	Required	62.0
appliedToId	String	ID of the invoice record to apply the credit for.	Required	62.0
description	String	Explanation or reason for applying the credit memo.	Optional	62.0
effectiveDate	String	Effective date for the credit memo.	Optional	62.0

Credit Memo Unapply Input

Input representation of the request to unapply a credit memo from an invoice.

JSON example

```
  {
    "description": "Unapply credit memo from invoice to revert an error",
    "effectiveDate": "2024-07-01"
  }
```

Properties

Name	Type	Description	Required or Optional	Available Version
description	String	Explanation or reason for unapplying the credit memo.	Optional	62.0
effectiveDate	String	Effective date for the credit memo.	Optional	62.0

Credit Memo Line Apply Input

Input representation of the details of the request to apply a credit memo line to an invoice line.

JSON example

```
{
 "applyCreditDetails": [
 {
 "invoiceLineId": "5TVSG0000002ZJR4A2",
 "appliedAmount": 5,
 "description": "Apply to invoice line 1",
 "effectiveDate": "2024-07-01"
 },
 {
 "invoiceLineId": "5TVSG0000002ZJS4A2",
 "appliedAmount": 10,
 "description": "Apply to invoice line 2",
 "effectiveDate": "2024-07-01"
 }
 ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
applyCreditDetails	Credit Memo Line Application Input[]	List of one or more applications to apply the credit memo line for. Each application represents an invoice line that's credited by using the balance of the specified credit memo line.	Required	62.0

Credit Memo Line Application Input

Input representation of the request to specify one or more applications to apply a credit memo line for, with each application representing an invoice line.

JSON example

```
 "applyCreditDetails": [
 {
 "invoiceLineId": "5TVSG0000002ZJR4A2",
 "appliedAmount": 5,
 "description": "Apply to invoice line 1",
 "effectiveDate": "2024-07-01"
 },
 {
 "invoiceLineId": "5TVSG0000002ZJS4A2",
 "appliedAmount": 10,
 "description": "Apply to invoice line 2",
 "effectiveDate": "2024-07-01"
```

```
 }
 ]
```

Properties

Name	Type	Description	Required or Optional	Available Version
appliedAmount	Double	Credit amount to be applied to the invoice line.	Required	62.0
description	String	Explanation or reason for applying the credit memo line.	Optional	62.0
effectiveDate	String	Effective date for the credit memo line.	Optional	62.0
invoiceLineId	String	ID of the invoice line record to apply the credit for.	Required	62.0

Credit Memo Line Unapply Input

Input representation of the details of the request to unapply a credit memo line from an invoice line.

JSON example

```
{
 "description": "Unapply a credit memo line from invoice line 1",
 "effectiveDate": "2024-07-01"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
description	String	Explanation or reason for unapplying the credit memo line.	Optional	62.0
effectiveDate	String	Effective date for the credit memo line.	Optional	62.0

Customer Details Input

Input representation of the customer details for tax calculation.

JSON example

```
{
    "accountId": "001R000000000zSMAQ"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`accountId`	String	Salesforce account ID of the customer.	Optional	62.0

Frequency Cadence Options

Input representation of the frequency cadence options for an invoice scheduler.

JSON example

```
"frequencyCadenceOptions": {
      "recurringSubType" : "Every",
      "recursOn" : "First",
      "recursOnDay" : "Sunday",
      "shouldExcludeWkendAndHldy": true
  }
```

Properties

Name	Type	Description	Required or Optional	Available Version
`recurringSubType`	String	Subtype of the recurring frequency for the invoice run. Valid values are: • `Every`—Specifies if the invoice scheduler must generate the invoices on a recurring frequency on a specific cadence. Use this value to generate invoices on a specific day of the month. For example, you can specify that the invoice scheduler must generate the invoices every first Monday of the month. • `SpecificDate`—Specifies if the invoice scheduler must generate the invoices on a recurring frequency on a specific date. Use this value to generate invoices on a monthly basis on a specific date.	Required if the `frequencyCadence` property is set to `Monthly`.	62.0
`recursOn`	String	Cadence that specifies when the invoice scheduler must generate the invoices on a recurring frequency. For example, you can specify that the invoice scheduler must generate the invoices every first Monday of the month. Valid values are: • `First`	Required if the `frequencyCadence` property is set to `Monthly`.	62.0

Name	Type	Description	Required or Optional	Available Version
		• `Second` • `Third` • `Fourth` • `Last`		
`recursOnDate`	String	Date when the invoice scheduler must generate the invoices on a specific date. The supported values are: • 1 through 28—Specify any date from 1 through 28. • `L`—Specifies that the invoice scheduler must generate the invoices on the last day of the month. • `L-1`—Specifies that the invoice scheduler must generate the invoices on the second to last day of the month. • `L-2`—Specifies that the invoice scheduler must generate the invoices on the third to last day of the month.	Required if the `recurringSubType` property is set to `SpecificDate`.	62.0
`recursOnDay`	String	Day of the week when the invoice scheduler must generate the invoices on a recurring frequency. For example, you can specify that the invoice scheduler must generate the invoices every Monday or every first Monday of a month. Valid values are: • `Sunday` • `Monday` • `Tuesday` • `Wednesday` • `Thursday` • `Friday` • `Saturday`	Required if the `frequencyCadence` property is set to `Weekly` or `Monthly`.	62.0
`shouldExcludeWkendAndHldy`	Boolean	Indicates whether to exclude weekends and holidays from the billing schedule (`true`) or not (`false`).	Required if the `frequencyCadence` property is set to `Monthly`.	62.0

Graph Record for Invoice Ingestion

A Graph record is an object that's a part of the graph structure, representing both the fields and relationships among different objects. Each record in the graph can contain attributes, which are fields of the object, and references to other related records.

Invoice ingestion supports a Graph record count of 500. The supported Graph attribute types for invoice ingestion are Account, Contact, Invoice, InvoiceLine, InvoiceLineTax, and InvoiceAddressGroup.

A Graph record has these properties and associated fields.

Field	Description
`referenceId`	Unique identifier of the record that's used to reference the record in the graph.
`record`	Object containing the actual data for the record.
`record.attributes.type`	Standard object of the record being referenced. In this scenario, this field indicates an `InvoiceAddressGroup` attribute type.
`record.attributes.method`	Method that defines the operation on the record. For example, POST to create a record, PUT to update a record, or GET to get record data.

JSON example

```
{
  "referenceId": "refBillingAddress",
  "record": {
    "attributes": {
      "type": "InvoiceAddressGroup",
      "method": "POST"
    },
    //Contains the actual data
  }
}
```

Account Record

Keep these considerations in mind when you specify an Account record as the Graph attribute type.

- The supported Graph attribute method is GET only.
- The supported record count that you can specify for Account is one only.
- The id field on attributes is the value of the externalId field API Name in Account. This field can be used to resolve the Salesforce Account ID from the custom external ID field. See Use an 'External ID' to set the values for audit fields.

JSON example

```
{
  "referenceId": "refAccount",
  "record": {
    "attributes": {
      "type": "Account",
      "method": "GET",
      "id": "AccountId__c/7661eaaf-f527-4dcf-beb3-301f3eddcd9e"
    }
```

```
  }
}
```

Contact Record

Keep these considerations in mind when you specify a Contact record as the Graph attribute type.

- The supported Graph attribute method is GET only.
- The supported record count that you can specify for Contact is one only.
- The id field on attributes is the value of the externalId field API Name in Contact. This field can be used to resolve the Salesforce Contact ID from the custom external ID field. See Use an 'External ID' to set the values for audit fields.

JSON example

```
{
  "referenceId": "refContact",
  "record": {
    "attributes": {
      "type": "Contact",
      "method": "GET",
      "id": "ContactId__c/7661eaaf-f527-4dcf-beb3-301f3eddcd9e"
    }
  }
}
```

Invoice Record

Keep these considerations in mind when you specify an Invoice record as the Graph attribute type.

- The supported Graph attribute method is POST only.
- The supported record count that you can specify for Invoice is one only.
- An invoice with `Draft` status can't have a posted date.
- An invoice with `Draft` status can't be used if the `taxCalculationStatus` property is set to `Posted`. The `taxCalculationStatus` property must be set to `Estimated` or `Pending`.
- An invoice with `Posted` status can't be used if the `shouldCalculateTax` property is set to `true`.
- An invoice with `Posted` status can't include the `Estimated` or `Pending` as the `taxCalculationStatus` property values. The tax processing status must be `Posted`.
- An invoice with `Posted` status with an invoice line marked as `taxable` must include an `invoiceLineTax` for that invoice line.
- An invoice with `Posted` status with an invoice line marked as `nontaxable` must not include an `invoiceLineTax` for that invoice line.
- The `invoice.CreationMode` field is set to `External`. This field differentiates the invoices created from a billing schedule or an invoice ingestion. For invoices created from a billing schedule, the value is Salesforce. For invoices created from an invoice ingestion, the value is set to `External`.
- The due date of an invoice graph must be set in the future or with a valid payment term, or your Salesforce org has a default payment term set.

JSON example

```
{
  "referenceId": "refInvoice",
```

```
  "record": {
    "attributes": {
      "type": "Invoice",
      "method": "POST"
    },
    "invoiceDate": "2024-01-01",
    "billingAccountId": "001xx000003Dzo9AAC",
    "billToContactId": "003xx000004TzFoAAI",
    "paymentTermId": "2OXxx0000004CFUGA2",
    "referenceEntityId": "801xx000003GeQQAA0",
    "status": "Draft",
    "currencyIsoCode": "USD",
    "dueDate": "2024-02-01",
    "postedDate": "2024-01-02",
    "invoiceNumber": "INV-12345",
    "uniqueIdentifier": "1a9380c1-8042-422d-bcc5-70c3f51c2588",
    "description": "Consulting Services"
  }
}
```

Table 1: Properties

NAME	Standard Object Field	Description	Required or Optional
`invoiceDate`	Invoice.invoiceDate	Date when the invoice was created or issued.	Required
`billingAccountId`	Invoice.billingAccountId	Billing account associated with this invoice, which can be resolved with the `externalId` field by using the GET method.	Required
`billToContactId`	Invoice.billToContactId	Contact person to whom the invoice is billed, which can be resolved with the `externalId` field by using the GET method.	Required
`paymentTermId`	Invoice.paymentTermId	Payment term associated with the invoice.	Optional
`referenceEntityId`	Invoice.referenceEntityId	Reference to a related object, if applicable.	Optional
`status`	Invoice.status	Current status of the invoice. Valid values are: • `Draft` • `Posted`. The default value is `Draft`.	Optional
`currencyIsoCode`	Invoice.currencyIsoCode	ISO code representing the currency of the invoice. This property must be specified if multi-currency is enabled in the organization.	Optional
`dueDate`	Invoice.dueDate	Due date for the invoice payment.	Optional
`postedDate`	Invoice.postedDate	Date the invoice was posted to the system.	Optional
`invoiceNumber`	Invoice.invoiceNumber	Unique identifier for the invoice.	Optional

NAME	Standard Object Field	Description	Required or Optional
uniqueIdentifier	Invoice.uniqueIdentifier	Unique identifier for the invoice. This property is used as the idempotency key to avoid duplicate invoice generation.	Optional
description	Invoice.description	Short description of the invoice or the billed items.	Optional

Invoice Line Record

Keep these considerations in mind when you specify an Invoice Line record as the Graph attribute type.

- The supported Graph attribute method is POST only.
- The Graph must include at least one Graph record with the InvoiceLine attribute type.
- The invoice line end date must be greater than or equal to the invoice line start date.

JSON example

```
{
  "referenceId": "refInvoiceLine",
  "record": {
    "attributes": {
      "type": "InvoiceLine",
      "method": "POST"
    },
    "name": "Parle G",
    "invoiceLineStartDate": "2024-01-01",
    "invoiceLineEndDate": "2024-01-31",
    "quantity": 10,
    "unitPrice": 25,
    "chargeAmount": 250,
    "invoiceId": "@{refInvoice.id}",
    "shippingAddressId": "@{refShippingAddress.id}",
    "billingAddressId": "@{refBillingAddress.id}",
    "referenceEntityItemId": "802xx000001neB9AAI",
    "taxTreatmentId": "1ttxx0000000BOTAA2",
    "legalEntityId": "0fwxx0000000001AAA",
    "legalEntityAccountingPeriodId": "1HLxx0000004C92GAE",
    "product2Id": "01txx0000006ic4AAA",
    "usageProductId": "01txx0000006ic5AAA",
    "isUsageBasedInvoiceLine": false,
    "usageOverageQuantity": 5,
    "unitOfMeasureId": "0hExx0000000001EAA",
    "description": "Sample product description"
  }
}
```

Table 2: Properties

NAME	Standard Object Field	Description	Required or Optional
name	InvoiceLine.name	Name or description of the invoice line item.	Required

NAME	Standard Object Field	Description	Required or Optional
`invoiceLineStartDate`	InvoiceLine.invoiceLineStartDate	Start date for the invoice line period. For example, the date when the product or service is provided.	Required
`invoiceLineEndDate`	InvoiceLine.invoiceLineEndDate	End date for the invoice line period. For example, the date when the product or service ends.	Required
`quantity`	InvoiceLine.quantity	Quantity of the product or billed service in this invoice line.	Required
`unitPrice`	InvoiceLine.unitPrice	Price per unit of the product or billed service.	Required
`chargeAmount`	InvoiceLine.chargeAmount	Total charge amount for this invoice line, which is calculated as quantity * unit price.	Required
`invoiceId`	InvoiceLine.invoiceId	Reference to the invoice that this invoice line is associated with.	Required
`shippingAddressId`	InvoiceLine.shippingAddressId	Shipping address associated with this invoice line.	Required
`billingAddressId`	InvoiceLine.billingAddressId	Billing address associated with this invoice line.	Required
`referenceEntityItemId`	InvoiceLine.referenceEntityItemId	Reference to a related object record for this invoice line.	Optional
`taxTreatmentId`	InvoiceLine.taxTreatmentId	Tax treatment applied to this invoice line such as taxable or nontaxable. If the `TaxTreatmentId` property isn't specified in the request, it's retrieved from the organization's default values. If the organization's defaults aren't set, an error is thrown.	Optional
`legalEntityId`	InvoiceLine.legalEntityId	Legal entity that this invoice line belongs to. If the `LegalEntityId` property isn't specified in the request, it's retrieved from the organization's default values. If the organization's defaults aren't set, an error is thrown.	Optional
`legalEntityAccountingPeriodId`	InvoiceLine.legalEntityAccountingPeriodId	Accounting period for the legal entity that this invoice line belongs to.	Optional
`product2Id`	InvoiceLine.product2Id	Identifier for the product that's billed in this invoice line.	Optional
`usageProductId`	InvoiceLine.usageProductId	Identifier for the usage-based product if this invoice line is for usage-based billing.	Optional
`isUsageBasedInvoiceLine`	InvoiceLine.isUsageBasedInvoiceLine	Boolean value that indicates whether this invoice line is for a usage-based product (`true`) or not (`false`).	Optional
`usageOverageQuantity`	InvoiceLine.usageOverageQuantity	Quantity of usage overage for this invoice line, if applicable.	Optional

NAME	Standard Object Field	Description	Required or Optional
`unitOfMeasureId`	InvoiceLine.unitOfMeasureId	Identifier for the unit of measure for the product or service, such as each, kg, or lb.	Optional
`description`	InvoiceLine.description	Description of the invoice line item with additional details.	Optional

Tax Record

Keep these considerations in mind when you specify a Tax record as the Graph attribute type.

- The supported Graph attribute method is POST only.
- The associated standard object is InvoiceLineTax.
- The invoiceLineTax graph record must not include the `taxCalculationStatus` property value as `Pending`.

JSON example

```
{
    "referenceId": "refInvoiceLineTax",
    "record": {
        "attributes": {
            "type": "InvoiceLineTax",
            "method": "POST"
        },
        "taxTransactionNumber": "TX123456789",
        "taxAmount": 15.00,
        "taxRate": 0.08,
        "taxName": "Sales Tax",
        "taxCode": "TX-SALES",
        "taxEffectiveDate": "2024-01-01",
        "invoiceLine": "@{refInvoiceLine.id}",
        "taxDocumentNumber": "TAXDOC987654",
        "taxExemptAmount": 0.00,
        "description": "Exempt from VAT"
    }
}
```

Table 3: Properties

NAME	Standard Object Field	Description	Required or Optional
`taxTransactionNumber`	InvoiceLineTax.taxTransactionNumber	Unique identifier of the tax transaction related to this invoice line tax.	Required
`taxAmount`	InvoiceLineTax.taxAmount	Total tax amount applied to the invoice line.	Required
`taxRate`	InvoiceLineTax.taxRate	Rate at which the tax is applied on the invoice line.	Required
`taxName`	InvoiceLineTax.taxName	Name of the tax applied on the invoice line, such as Sales Tax or Value Added Tax (VAT).	Required
`taxCode`	InvoiceLineTax.taxCode	Tax code that corresponds to the applied tax rate.	Required

NAME	Standard Object Field	Description	Required or Optional
`taxEffectiveDate`	InvoiceLineTax.taxEffectiveDate	Effective date from which the tax rate applies.	Required
`invoiceLine`	InvoiceLineTax.invoiceLine	Reference to the invoice line this tax record is associated with.	Required
`taxDocumentNumber`	InvoiceLineTax.taxDocumentNumber	Document number associated with the tax transaction, such as tax return document or government filing number.	Required
`taxExemptAmount`	InvoiceLine.billingAddressId	Billing address associated with this invoice line.	Required
`description`	InvoiceLineTax.description	Additional details about the tax.	Optional

Shipping Address Record

Keep these considerations in mind when you specify a Shipping Address record as the Graph attribute type.

- The supported Graph attribute method is POST only.
- The associated standard object is InvoiceAddressGroup.

JSON example

```
{
    "referenceId": "refShippingAddress",
    "record": {
        "attributes": {
            "type": "InvoiceAddressGroup",
            "method": "POST"
        },
        "street": "456 Elm St",
        "city": "Springfield",
        "postalCode": "62701",
        "state": "IL",
        "country": "USA",
        "longitude": "-89.6501",
        "latitude": "39.7817",
        "invoiceId": "@{refInvoice.id}"
    }
}
```

Table 4: Properties

NAME	Standard Object Field	Description	Required or Optional
`invoiceId`	InvoiceAddressGroup.invoice	Unique identifier of the invoice associated with this shipping address.	Required
`street`	InvoiceAddressGroup.street	Street address where the goods or services are shipped to.	Required
`city`	InvoiceAddressGroup.city	City where the shipment is delivered.	Required

NAME	Standard Object Field	Description	Required or Optional
postalCode	InvoiceAddressGroup.postalCode	Postal or ZIP code for the shipping address.	Required
state	InvoiceAddressGroup.state	State or province of the delivery location.	Required
country	InvoiceAddressGroup.country	Country where the shipment is delivered.	Required
longitude	InvoiceAddressGroup.longitude	Geographic longitude of the shipping address location for mapping purposes.	Optional
latitude	InvoiceAddressGroup.latitude	Geographic latitude of the shipping address location for mapping purposes.	Optional

Billing Address Record

Keep these considerations in mind when you specify a Billing Address record as the Graph attribute type.

- The supported Graph attribute method is POST only.
- The associated standard object is InvoiceAddressGroup.

JSON example

```
{
    "referenceId": "refBillingAddress",
    "record": {
        "attributes": {
            "type": "InvoiceAddressGroup",
            "method": "POST"
        },
        "street": "789 Oak St",
        "city": "Los Angeles",
        "postalCode": "90001",
        "state": "CA",
        "country": "USA",
        "longitude": "-118.2437",
        "latitude": "34.0522",
        "invoiceId": "@{refInvoice.id}"
    }
}
```

Table 5: Properties

NAME	Standard Object Field	Description	Required or Optional
invoiceId	InvoiceAddressGroup.invoice	Unique identifier of the invoice associated with this billing address.	Required
street	InvoiceAddressGroup.street	Street address for billing purposes.	Required
city	InvoiceAddressGroup.city	City for billing address.	Required
postalCode	InvoiceAddressGroup.postalCode	Postal or ZIP code for the billing address.	Required
state	InvoiceAddressGroup.state	State or province for the billing address.	Required

NAME	Standard Object Field	Description	Required or Optional
`country`	InvoiceAddressGroup.country	Country for the billing address.	Required
`longitude`	InvoiceAddressGroup.longitude	Geographic longitude of the billing address location for mapping purposes.	Optional
`latitude`	InvoiceAddressGroup.latitude	Geographic latitude of the billing address location for mapping purposes.	Optional

Invoice Draft To Posted Input

Input representation of the details of the draft invoice that's posted.

JSON example

```
{
  "invoiceIds": ["3ttxx0000004CIjAAM"]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`correlationId`	String	Splunk correlation ID to track the messages that are related to the request and are logged in Splunk by the different services involved in the request. If the ID isn't specified, the service creates a random Universally Unique Identifier (UUID).	Optional	62.0
`invoiceIds`	String[]	IDs of the invoice records in `Draft` status to be posted. You can post one draft invoice per API request.	Required	62.0

Invoice Estimated Tax Calculation Input

Details of the invoice for which the estimated tadsfsefx must be calculated.

JSON example

```
{
  "invoiceIds": ["3ttxx0000004CIjAAM"]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`correlationId`	String	Splunk correlation ID to track the messages that are related to the request and are logged in Splunk by the different services involved in the request. If the ID isn't specified, the service creates a random Universally Unique Identifier (UUID).	Optional	63.0
`invoiceIds`	String[]	IDs of the invoices for which the estimated tax must be calculated. You can specify one invoice per API request.	Required	63.0

Invoice Ingestion Input

Input representation of the details of the invoice to be processed. The details include the tax processing status, user preferences for tax callouts, and associated object graph representation.

JSON example

This example shows a sample request to ingest a draft invoice with a tax callout.

```
{
  "invoices": [
    {
      "shouldCalculateTax": true,
      "taxCalculationStatus": "Estimated",
      "graph": {
        "graphId": "CreateInvoice",
        "records": [
          {
            "referenceId": "refAccount",
            "record": {
              "attributes": {
                "type": "Account",
                "method": "GET",
                "id": "ExternalId__c/123"
              }
            }
          },
          {
            "referenceId": "refContact",
            "record": {
              "attributes": {
                "type": "Contact",
                "method": "GET",
                "id": "ExternalId__c/123"
              }
            }
          },
          {
```

```
      "referenceId": "refInvoice",
      "record": {
        "Attributes": {
          "type": "Invoice",
          "method": "POST"
        },
        "billingAccountId": "001SG00000mKFfGYAW",
        "billToContactId": "003SG00000Bnv5HYAR",
        "paymentTermId": "2OXxx0000004CFUGA2",
        "referenceEntityId": "801xx000003GeQQAA0",
        "status": "Draft",
        "invoiceDate": "2024-12-19",
        "currencyIsoCode": "USD",
        "dueDate": "2024-12-19",
        "invoiceNumber": "DOC-10",
        "description": "Sample Invoice",
        "uniqueIdentifier": "5873af8f-f007-4aa0-9e3d-53a08c3f59de"
      }
    },
    {
      "referenceId": "refBillingAddress",
      "record": {
        "Attributes": {
          "type": "InvoiceAddressGroup",
          "method": "POST"
        },
        "street": "123 Main St",
        "city": "NewYork",
        "postalCode": "10001",
        "state": "New York",
        "country": "US",
        "longitude": "123.456",
        "latitude": "78.910",
        "invoiceId": "3ttSG000000EPjJYAW"
      }
    },
    {
      "referenceId": "refShippingAddress",
      "record": {
        "attributes": {
          "type": "InvoiceAddressGroup",
          "method": "POST"
        },
        "street": "123 Main St",
        "city": "NewYork",
        "postalCode": "10001",
        "state": "New York",
        "country": "US",
        "longitude": "123.456",
        "latitude": "78.910",
        "invoiceId": "3ttSG000000EPjJYAW"
      }
    },
    {
```

```
      "referenceId": "refInvoiceLine1",
      "record": {
        "attributes": {
          "type": "InvoiceLine",
          "method": "POST"
        },
        "name": "productName1",
        "product2Id": "01txx0000006ic4AAA",
        "invoiceLineStartDate": "2024-11-10",
        "invoiceLineEndDate": "2024-11-13",
        "quantity": "10",
        "unitPrice": "10",
        "chargeAmount": "100",
        "invoiceId": "3ttSG000000EPjJYAW",
        "referenceEntityItemId": "802xx000001neB9AAI",
        "billingAddressId": "8zzSG0000000IbFYAU",
        "shippingAddressId": "8zzSG0000000IbGYAU",
        "taxTreatmentId": "1ttxx0000000BOTAA2",
        "legalEntityId": "0fwxx0000000001AAA",
        "legalEntityAccountingPeriodId": "1HLxx0000004C92GAE"
      }
    },
    {
      "referenceId": "refInvoiceLine2",
      "record": {
        "attributes": {
          "type": "InvoiceLine",
          "method": "POST"
        },
        "name": "productName2",
        "product2Id": "01txx0000006ic4AAA",
        "invoiceLineStartDate": "2024-11-10",
        "invoiceLineEndDate": "2024-11-15",
        "quantity": "10",
        "unitPrice": "10",
        "chargeAmount": "100",
        "invoiceId": "3ttSG000000EPjJYAW",
        "referenceEntityItemId": "802xx000001neB9AAI",
        "billingAddressId": "8zzSG0000000IbFYAU",
        "shippingAddressId": "8zzSG0000000IbGYAU",
        "taxTreatmentId": "1ttxx00000001DpAAI",
        "legalEntityId": "0fwxx0000000001AAA",
        "legalEntityAccountingPeriodId": "1HLxx0000004C92GAE"
      }
    },
    {
      "referenceId": "refInvoiceLineTax",
      "record": {
        "attributes": {
          "type": "InvoiceLineTax",
          "method": "POST"
        },
        "taxAmount": 7.25,
        "taxCode": "CA-94121",
```

```
              "taxName": "CALIFORNIA",
              "taxRate": 0.25,
              "taxEffectiveDate": "2024-11-10",
              "taxDocumentNumber": "123",
              "taxExemptAmount": 0,
              "taxTransactionNumber": "kl",
              "description": "Associated tax line.",
              "invoiceLineId": "5TVSG0000004dAr4AI"
            }
          },
          {
            "referenceId": "refInvoiceLineTax1",
            "record": {
              "attributes": {
                "type": "InvoiceLineTax",
                "method": "POST"
              },
              "taxAmount": "10",
              "taxCode": "CA-94121",
              "taxName": "CALIFORNIA",
              "taxRate": 0.25,
              "taxEffectiveDate": "2024-11-10",
              "taxDocumentNumber": "123",
              "taxExemptAmount": 0,
              "taxTransactionNumber": "125",
              "invoiceLineId": "5TVSG0000004xeD4AQ",
              "description": "Associated tax line."
            }
          }
        ]
      }
    }
  ]
}
```

This example shows a sample request ingest a draft invoice without a tax callout.

```
{
  "invoices": [
    {
      "taxCalculationStatus": "Estimated",
      "graph": {
        "graphId": "123",
        "records": [
          {
            "referenceId": "refAccount",
            "record": {
              "attributes": {
                "type": "Account",
                "method": "GET",
                "id": "TestExternalId__c/123"
              }
            }
          },
          {
```

```
      "referenceId": "refInvoice",
      "record": {
        "Attributes": {
          "type": "Invoice",
          "method": "POST"
        },
        "billingAccountId": "@{refAccount.Id}",
        "billToContactId": "003xx000004Wk8qAAC",
        "paymentTermId": "2OXxx0000004CFUGA2",
        "referenceEntityId": "801xx000003GeQQAA0",
        "status": "Draft",
        "invoiceDate": "2024-12-19",
        "currencyIsoCode": "USD",
        "dueDate": "2024-12-19",
        "invoiceNumber": "DOC-10",
        "description": "testInvoice",
        "uniqueIdentifier": "c76011a1-e113-49d9-9c54-3b5c68950ada"
      }
    },
    {
      "referenceId": "refBillingAddress",
      "record": {
        "Attributes": {
          "type": "InvoiceAddressGroup",
          "method": "POST"
        },
        "street": "123 Main St",
        "city": "NewYork",
        "postalCode": "10001",
        "state": "New York",
        "country": "US",
        "longitude": "123.456",
        "latitude": "78.910",
        "invoiceId": "@{refInvoice.id}"
      }
    },
    {
      "referenceId": "refShippingAddress",
      "record": {
        "attributes": {
          "type": "InvoiceAddressGroup",
          "method": "POST"
        },
        "street": "123 Main St",
        "city": "NewYork",
        "postalCode": "10001",
        "state": "New York",
        "country": "US",
        "longitude": "123.456",
        "latitude": "78.910",
        "invoiceId": "@{refInvoice.id}"
      }
    },
    {
```

```
      "referenceId": "refInvoiceLine1",
      "record": {
        "attributes": {
          "type": "InvoiceLine",
          "method": "POST"
        },
        "name": "productName1",
        "product2Id": "01txx0000006ic4AAA",
        "invoiceLineStartDate": "2024-11-10",
        "invoiceLineEndDate": "2024-11-13",
        "quantity": "10",
        "unitPrice": "10",
        "chargeAmount": "100",
        "invoiceId": "@{refInvoice.id}",
        "referenceEntityItemId": "802xx000001neB9AAI",
        "billingAddressId": "@{refBillingAddress.id}",
        "shippingAddressId": "@{refShippingAddress.id}",
        "taxTreatmentId": "1ttxx0000000BOTAA2",
        "legalEntityId": "0fwxx0000000001AAA",
        "legalEntityAccountingPeriodId": "1HLxx0000004C92GAE"
      }
    },
    {
      "referenceId": "refInvoiceLine2",
      "record": {
        "attributes": {
          "type": "InvoiceLine",
          "method": "POST"
        },
        "name": "productName2",
        "product2Id": "01txx0000006ic4AAA",
        "invoiceLineStartDate": "2024-11-10",
        "invoiceLineEndDate": "2024-11-15",
        "quantity": "10",
        "unitPrice": "10",
        "chargeAmount": "100",
        "invoiceId": "@{refInvoice.id}",
        "referenceEntityItemId": "802xx000001neB9AAI",
        "billingAddressId": "@{refBillingAddress.id}",
        "shippingAddressId": "@{refShippingAddress.id}",
        "taxTreatmentId": "1ttxx00000001DpAAI",
        "legalEntityId": "0fwxx0000000001AAA",
        "legalEntityAccountingPeriodId": "1HLxx0000004C92GAE"
      }
    },
    {
      "referenceId": "refInvoiceLineTax",
      "record": {
        "attributes": {
          "type": "InvoiceLineTax",
          "method": "POST"
        },
        "taxAmount": 7.25,
        "taxCode": "CA-94121",
```

```
              "taxName": "CALIFORNIA",
              "taxRate": 0.25,
              "taxEffectiveDate": "2024-11-10",
              "taxDocumentNumber": "123",
              "taxExemptAmount": 0,
              "taxTransactionNumber": "kl",
              "description": "description for tax Line",
              "invoiceLineId": "@{refInvoiceLine1.id}"
            }
          },
          {
            "referenceId": "refInvoiceLineTax1",
            "record": {
              "attributes": {
                "type": "InvoiceLineTax",
                "method": "POST"
              },
              "taxAmount": "10",
              "taxCode": "CA-94121",
              "taxName": "CALIFORNIA",
              "taxRate": 0.25,
              "taxEffectiveDate": "2024-11-10",
              "taxDocumentNumber": "123",
              "taxExemptAmount": 0,
              "taxTransactionNumber": "125",
              "invoiceLineId": "@{refInvoiceLine2.id}",
              "description": "description for tax Line"
            }
          }
        ]
      }
    }
  ]
}
```

This example shows a sample request to ingest posted invoices.

```
{
  "invoices": [
    {
      "taxCalculationStatus": "Posted",
      "graph": {
        "graphId": "123",
        "records": [
          {
            "referenceId": "refAccount",
            "record": {
              "attributes": {
                "type": "Account",
                "method": "GET",
                "id": "ExternalId__c/123"
              }
            }
          },
          {
```

```
      "referenceId": "refInvoice",
      "record": {
        "Attributes": {
          "type": "Invoice",
          "method": "POST"
        },
        "billingAccountId": "001SG00000njpF3YAI",
        "billToContactId": "003xx000004Wk8qAAC",
        "paymentTermId": "2OXxx0000004CFUGA2",
        "referenceEntityId": "801xx000003GeQQAA0",
        "status": "Posted",
        "invoiceDate": "2024-12-19",
        "currencyIsoCode": "USD",
        "dueDate": "2024-12-19",
        "invoiceNumber": "DOC-10",
        "description": "Sample Invoice",
        "uniqueIdentifier": "9994b2c4-c0c3-47c3-806f-ae6e1f16bac3"
      }
    },
    {
      "referenceId": "refBillingAddress",
      "record": {
        "Attributes": {
          "type": "InvoiceAddressGroup",
          "method": "POST"
        },
        "street": "123 Main St",
        "city": "NewYork",
        "postalCode": "10001",
        "state": "New York",
        "country": "US",
        "longitude": "123.456",
        "latitude": "78.910",
        "invoiceId": "3ttSG000000FFWbYAO"
      }
    },
    {
      "referenceId": "refShippingAddress",
      "record": {
        "attributes": {
          "type": "InvoiceAddressGroup",
          "method": "POST"
        },
        "street": "123 Main St",
        "city": "NewYork",
        "postalCode": "10001",
        "state": "New York",
        "country": "US",
        "longitude": "123.456",
        "latitude": "78.910",
        "invoiceId": "3ttSG000000FFWbYAO"
      }
    },
    {
```

```
      "referenceId": "refInvoiceLine1",
      "record": {
        "attributes": {
          "type": "InvoiceLine",
          "method": "POST"
        },
        "name": "productName1",
        "product2Id": "01txx0000006ic4AAA",
        "invoiceLineStartDate": "2024-11-10",
        "invoiceLineEndDate": "2024-11-13",
        "quantity": "10",
        "unitPrice": "10",
        "chargeAmount": "100",
        "invoiceId": "3ttSG000000FFWbYAO",
        "referenceEntityItemId": "802xx000001neB9AAI",
        "billingAddressId": "8zzSG0000000NmDYAU",
        "shippingAddressId": "8zzSG0000000IbGYAU",
        "taxTreatmentId": "1ttxx0000000BOTAA2",
        "legalEntityId": "0fwxx0000000001AAA",
        "legalEntityAccountingPeriodId": "1HLxx0000004C92GAE"
      }
    },
    {
      "referenceId": "refInvoiceLine2",
      "record": {
        "attributes": {
          "type": "InvoiceLine",
          "method": "POST"
        },
        "name": "productName2",
        "product2Id": "01txx0000006ic4AAA",
        "invoiceLineStartDate": "2024-11-10",
        "invoiceLineEndDate": "2024-11-15",
        "quantity": "10",
        "unitPrice": "10",
        "chargeAmount": "100",
        "invoiceId": "3ttSG000000FFWbYAO",
        "referenceEntityItemId": "802xx000001neB9AAI",
        "billingAddressId": "8zzSG0000000NmDYAU",
        "shippingAddressId": "8zzSG0000000IbGYAU",
        "taxTreatmentId": "1ttxx00000001DpAAI",
        "legalEntityId": "0fwxx0000000001AAA",
        "legalEntityAccountingPeriodId": "1HLxx0000004C92GAE"
      }
    },
    {
      "referenceId": "refInvoiceLineTax1",
      "record": {
        "attributes": {
          "type": "InvoiceLineTax",
          "method": "POST"
        },
        "taxAmount": "10",
        "taxCode": "CA-94121",
```

```
                "taxName": "CALIFORNIA",
                "taxRate": 0.25,
                "taxEffectiveDate": "2024-11-10",
                "taxDocumentNumber": "123",
                "taxExemptAmount": 0,
                "taxTransactionNumber": "125",
                "invoiceLineId": "5TVSG0000004mJ04AI",
                "description": "Associated tax line."
            }
        }
    ]
  }
}
}
]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`correlationId`	String	Splunk correlation ID to track the messages that are related to the request and are logged in Splunk by the different services involved in the request. If the ID isn't specified, the service creates a random Universally Unique Identifier (UUID).	Optional	63.0
`graph`	Object Graph Input	Graph that represents the invoice structure for invoice ingestion or generation. The supported Graph attribute types for invoice ingestion are Account, Contact, Invoice, InvoiceLine, InvoiceLineTax, and InvoiceAddressGroup. See Graph Record for Invoice Ingestion.	Required	63.0
`shouldCalculateTax`	Boolean	Indicates whether the estimated tax must be calculated for the ingested invoice (`true`) or not (`false`). The default value is `false`.	Optional	63.0
`taxCalculationStatus`	String	Status of the tax calculation, which is saved on the invoice line. Valid values are: • `Estimated` • `Pending` • `Posted` The default value is `Pending`.	Optional	63.0

Invoice Input for Ingestion

Input representation of the details of the invoice that must be generated for or ingested into Billing.

JSON example

This example shows a sample request to ingest a draft invoice with a tax callout.

```
{
  "invoices": [
    {
      "shouldCalculateTax": true,
      "taxCalculationStatus": "Estimated",
      "graph": {
        "graphId": "CreateInvoice",
        "records": [
          {
            "referenceId": "refAccount",
            "record": {
              "attributes": {
                "type": "Account",
                "method": "GET",
                "id": "ExternalId__c/123"
              }
            }
          },
          {
            "referenceId": "refContact",
            "record": {
              "attributes": {
                "type": "Contact",
                "method": "GET",
                "id": "ExternalId__c/123"
              }
            }
          },
          {
            "referenceId": "refInvoice",
            "record": {
              "Attributes": {
                "type": "Invoice",
                "method": "POST"
              },
              "billingAccountId": "001SG00000mKFfGYAW",
              "billToContactId": "003SG00000Bnv5HYAR",
              "paymentTermId": "2OXxx0000004CFUGA2",
              "referenceEntityId": "801xx000003GeQQAA0",
              "status": "Draft",
              "invoiceDate": "2024-12-19",
              "currencyIsoCode": "USD",
              "dueDate": "2024-12-19",
              "invoiceNumber": "DOC-10",
              "description": "Sample Invoice",
              "uniqueIdentifier": "5873af8f-f007-4aa0-9e3d-53a08c3f59de"
            }
          },
```

```
{
  "referenceId": "refBillingAddress",
  "record": {
    "Attributes": {
      "type": "InvoiceAddressGroup",
      "method": "POST"
    },
    "street": "123 Main St",
    "city": "NewYork",
    "postalCode": "10001",
    "state": "New York",
    "country": "US",
    "longitude": "123.456",
    "latitude": "78.910",
    "invoiceId": "3ttSG000000EPjJYAW"
  }
},
{
  "referenceId": "refShippingAddress",
  "record": {
    "attributes": {
      "type": "InvoiceAddressGroup",
      "method": "POST"
    },
    "street": "123 Main St",
    "city": "NewYork",
    "postalCode": "10001",
    "state": "New York",
    "country": "US",
    "longitude": "123.456",
    "latitude": "78.910",
    "invoiceId": "3ttSG000000EPjJYAW"
  }
},
{
  "referenceId": "refInvoiceLine1",
  "record": {
    "attributes": {
      "type": "InvoiceLine",
      "method": "POST"
    },
    "name": "productName1",
    "product2Id": "01txx0000006ic4AAA",
    "invoiceLineStartDate": "2024-11-10",
    "invoiceLineEndDate": "2024-11-13",
    "quantity": "10",
    "unitPrice": "10",
    "chargeAmount": "100",
    "invoiceId": "3ttSG000000EPjJYAW",
    "referenceEntityItemId": "802xx000001neB9AAI",
    "billingAddressId": "8zzSG0000000IbFYAU",
    "shippingAddressId": "8zzSG0000000IbGYAU",
    "taxTreatmentId": "1ttxx0000000BOTAA2",
    "legalEntityId": "0fwxx0000000001AAA",
```

```
        "legalEntityAccountingPeriodId": "1HLxx0000004C92GAE"
      }
    },
    {
      "referenceId": "refInvoiceLine2",
      "record": {
        "attributes": {
          "type": "InvoiceLine",
          "method": "POST"
        },
        "name": "productName2",
        "product2Id": "01txx0000006ic4AAA",
        "invoiceLineStartDate": "2024-11-10",
        "invoiceLineEndDate": "2024-11-15",
        "quantity": "10",
        "unitPrice": "10",
        "chargeAmount": "100",
        "invoiceId": "3ttSG000000EPjJYAW",
        "referenceEntityItemId": "802xx000001neB9AAI",
        "billingAddressId": "8zzSG0000000IbFYAU",
        "shippingAddressId": "8zzSG0000000IbGYAU",
        "taxTreatmentId": "1ttxx00000001DpAAI",
        "legalEntityId": "0fwxx0000000001AAA",
        "legalEntityAccountingPeriodId": "1HLxx0000004C92GAE"
      }
    },
    {
      "referenceId": "refInvoiceLineTax",
      "record": {
        "attributes": {
          "type": "InvoiceLineTax",
          "method": "POST"
        },
        "taxAmount": 7.25,
        "taxCode": "CA-94121",
        "taxName": "CALIFORNIA",
        "taxRate": 0.25,
        "taxEffectiveDate": "2024-11-10",
        "taxDocumentNumber": "123",
        "taxExemptAmount": 0,
        "taxTransactionNumber": "kl",
        "description": "Associated tax line.",
        "invoiceLineId": "5TVSG0000004dAr4AI"
      }
    },
    {
      "referenceId": "refInvoiceLineTax1",
      "record": {
        "attributes": {
          "type": "InvoiceLineTax",
          "method": "POST"
        },
        "taxAmount": "10",
        "taxCode": "CA-94121",
```

```
              "taxName": "CALIFORNIA",
              "taxRate": 0.25,
              "taxEffectiveDate": "2024-11-10",
              "taxDocumentNumber": "123",
              "taxExemptAmount": 0,
              "taxTransactionNumber": "125",
              "invoiceLineId": "5TVSG0000004xeD4AQ",
              "description": "Associated tax line."
            }
          }
        ]
      }
    }
  ]
}
```

This example shows a sample request ingest a draft invoice without a tax callout.

```
{
  "invoices": [
    {
      "taxCalculationStatus": "Estimated",
      "graph": {
        "graphId": "123",
        "records": [
          {
            "referenceId": "refAccount",
            "record": {
              "attributes": {
                "type": "Account",
                "method": "GET",
                "id": "TestExternalId__c/123"
              }
            }
          },
          {
            "referenceId": "refInvoice",
            "record": {
              "Attributes": {
                "type": "Invoice",
                "method": "POST"
              },
              "billingAccountId": "@{refAccount.Id}",
              "billToContactId": "003xx000004Wk8qAAC",
              "paymentTermId": "2OXxx0000004CFUGA2",
              "referenceEntityId": "801xx000003GeQQAA0",
              "status": "Draft",
              "invoiceDate": "2024-12-19",
              "currencyIsoCode": "USD",
              "dueDate": "2024-12-19",
              "invoiceNumber": "DOC-10",
              "description": "testInvoice",
              "uniqueIdentifier": "c76011a1-e113-49d9-9c54-3b5c68950ada"
            }
          },
```

```
{
  "referenceId": "refBillingAddress",
  "record": {
    "Attributes": {
      "type": "InvoiceAddressGroup",
      "method": "POST"
    },
    "street": "123 Main St",
    "city": "NewYork",
    "postalCode": "10001",
    "state": "New York",
    "country": "US",
    "longitude": "123.456",
    "latitude": "78.910",
    "invoiceId": "@{refInvoice.id}"
  }
},
{
  "referenceId": "refShippingAddress",
  "record": {
    "attributes": {
      "type": "InvoiceAddressGroup",
      "method": "POST"
    },
    "street": "123 Main St",
    "city": "NewYork",
    "postalCode": "10001",
    "state": "New York",
    "country": "US",
    "longitude": "123.456",
    "latitude": "78.910",
    "invoiceId": "@{refInvoice.id}"
  }
},
{
  "referenceId": "refInvoiceLine1",
  "record": {
    "attributes": {
      "type": "InvoiceLine",
      "method": "POST"
    },
    "name": "productName1",
    "product2Id": "01txx0000006ic4AAA",
    "invoiceLineStartDate": "2024-11-10",
    "invoiceLineEndDate": "2024-11-13",
    "quantity": "10",
    "unitPrice": "10",
    "chargeAmount": "100",
    "invoiceId": "@{refInvoice.id}",
    "referenceEntityItemId": "802xx000001neB9AAI",
    "billingAddressId": "@{refBillingAddress.id}",
    "shippingAddressId": "@{refShippingAddress.id}",
    "taxTreatmentId": "1ttxx0000000BOTAA2",
    "legalEntityId": "0fwxx0000000001AAA",
```

```
        "legalEntityAccountingPeriodId": "1HLxx0000004C92GAE"
      }
    },
    {
      "referenceId": "refInvoiceLine2",
      "record": {
        "attributes": {
          "type": "InvoiceLine",
          "method": "POST"
        },
        "name": "productName2",
        "product2Id": "01txx0000006ic4AAA",
        "invoiceLineStartDate": "2024-11-10",
        "invoiceLineEndDate": "2024-11-15",
        "quantity": "10",
        "unitPrice": "10",
        "chargeAmount": "100",
        "invoiceId": "@{refInvoice.id}",
        "referenceEntityItemId": "802xx000001neB9AAI",
        "billingAddressId": "@{refBillingAddress.id}",
        "shippingAddressId": "@{refShippingAddress.id}",
        "taxTreatmentId": "1ttxx00000001DpAAI",
        "legalEntityId": "0fwxx0000000001AAA",
        "legalEntityAccountingPeriodId": "1HLxx0000004C92GAE"
      }
    },
    {
      "referenceId": "refInvoiceLineTax",
      "record": {
        "attributes": {
          "type": "InvoiceLineTax",
          "method": "POST"
        },
        "taxAmount": 7.25,
        "taxCode": "CA-94121",
        "taxName": "CALIFORNIA",
        "taxRate": 0.25,
        "taxEffectiveDate": "2024-11-10",
        "taxDocumentNumber": "123",
        "taxExemptAmount": 0,
        "taxTransactionNumber": "kl",
        "description": "description for tax Line",
        "invoiceLineId": "@{refInvoiceLine1.id}"
      }
    },
    {
      "referenceId": "refInvoiceLineTax1",
      "record": {
        "attributes": {
          "type": "InvoiceLineTax",
          "method": "POST"
        },
        "taxAmount": "10",
        "taxCode": "CA-94121",
```

```
              "taxName": "CALIFORNIA",
              "taxRate": 0.25,
              "taxEffectiveDate": "2024-11-10",
              "taxDocumentNumber": "123",
              "taxExemptAmount": 0,
              "taxTransactionNumber": "125",
              "invoiceLineId": "@{refInvoiceLine2.id}",
              "description": "description for tax Line"
            }
          }
        ]
      }
    }
  ]
}
```

This example shows a sample request to ingest posted invoices.

```
{
  "invoices": [
    {
      "taxCalculationStatus": "Posted",
      "graph": {
        "graphId": "123",
        "records": [
          {
            "referenceId": "refAccount",
            "record": {
              "attributes": {
                "type": "Account",
                "method": "GET",
                "id": "ExternalId__c/123"
              }
            }
          },
          {
            "referenceId": "refInvoice",
            "record": {
              "Attributes": {
                "type": "Invoice",
                "method": "POST"
              },
              "billingAccountId": "001SG00000njpF3YAI",
              "billToContactId": "003xx000004Wk8qAAC",
              "paymentTermId": "2OXxx0000004CFUGA2",
              "referenceEntityId": "801xx000003GeQQAA0",
              "status": "Posted",
              "invoiceDate": "2024-12-19",
              "currencyIsoCode": "USD",
              "dueDate": "2024-12-19",
              "invoiceNumber": "DOC-10",
              "description": "Sample Invoice",
              "uniqueIdentifier": "9994b2c4-c0c3-47c3-806f-ae6e1f16bac3"
            }
          },
```

```
        {
          "referenceId": "refBillingAddress",
          "record": {
            "Attributes": {
              "type": "InvoiceAddressGroup",
              "method": "POST"
            },
            "street": "123 Main St",
            "city": "NewYork",
            "postalCode": "10001",
            "state": "New York",
            "country": "US",
            "longitude": "123.456",
            "latitude": "78.910",
            "invoiceId": "3ttSG000000FFWbYAO"
          }
        },
        {
          "referenceId": "refShippingAddress",
          "record": {
            "attributes": {
              "type": "InvoiceAddressGroup",
              "method": "POST"
            },
            "street": "123 Main St",
            "city": "NewYork",
            "postalCode": "10001",
            "state": "New York",
            "country": "US",
            "longitude": "123.456",
            "latitude": "78.910",
            "invoiceId": "3ttSG000000FFWbYAO"
          }
        },
        {
          "referenceId": "refInvoiceLine1",
          "record": {
            "attributes": {
              "type": "InvoiceLine",
              "method": "POST"
            },
            "name": "productName1",
            "product2Id": "01txx0000006ic4AAA",
            "invoiceLineStartDate": "2024-11-10",
            "invoiceLineEndDate": "2024-11-13",
            "quantity": "10",
            "unitPrice": "10",
            "chargeAmount": "100",
            "invoiceId": "3ttSG000000FFWbYAO",
            "referenceEntityItemId": "802xx000001neB9AAI",
            "billingAddressId": "8zzSG0000000NmDYAU",
            "shippingAddressId": "8zzSG0000000IbGYAU",
            "taxTreatmentId": "1ttxx0000000BOTAA2",
            "legalEntityId": "0fwxx0000000001AAA",
```

```
              "legalEntityAccountingPeriodId": "1HLxx0000004C92GAE"
            }
          },
          {
            "referenceId": "refInvoiceLine2",
            "record": {
              "attributes": {
                "type": "InvoiceLine",
                "method": "POST"
              },
              "name": "productName2",
              "product2Id": "01txx0000006ic4AAA",
              "invoiceLineStartDate": "2024-11-10",
              "invoiceLineEndDate": "2024-11-15",
              "quantity": "10",
              "unitPrice": "10",
              "chargeAmount": "100",
              "invoiceId": "3ttSG000000FFWbYAO",
              "referenceEntityItemId": "802xx000001neB9AAI",
              "billingAddressId": "8zzSG0000000NmDYAU",
              "shippingAddressId": "8zzSG0000000IbGYAU",
              "taxTreatmentId": "1ttxx00000001DpAAI",
              "legalEntityId": "0fwxx0000000001AAA",
              "legalEntityAccountingPeriodId": "1HLxx0000004C92GAE"
            }
          },
          {
            "referenceId": "refInvoiceLineTax1",
            "record": {
              "attributes": {
                "type": "InvoiceLineTax",
                "method": "POST"
              },
              "taxAmount": "10",
              "taxCode": "CA-94121",
              "taxName": "CALIFORNIA",
              "taxRate": 0.25,
              "taxEffectiveDate": "2024-11-10",
              "taxDocumentNumber": "123",
              "taxExemptAmount": 0,
              "taxTransactionNumber": "125",
              "invoiceLineId": "5TVSG0000004mJ04AI",
              "description": "Associated tax line."
            }
          }
        ]
      }
    }
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`invoices`	Invoice Ingestion Input[]	List of invoices to be generated or ingested, enabling the submission and processing of multiple invoices in a single request. This API supports one invoice per request. To send 25 requests at a time, see the Composite Batch request.	Required	63.0

Invoice Input

Input representation of the details of the billing schedule.

JSON example

```
{
  "accountId": "001SG00000mYtRWYA0",
  "action": "Posted",
  "billingScheduleIds": [
    "44bSG000000CVeMYAW"
  ],
  "billingTransactionId": "801SG00000mYtaXYAS",
  "correlationId": null,
  "invoiceDate": "2024-01-12",
  "targetDate": "2024-01-12"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`accountId`	String	ID of the account record to create the invoices for.	Required if the `billingScheduleIds` or `billingTransactionId` property isn't specified.	63.0
`action`	String	Type of invoice to be created. Valid values are: • `Draft` • `Posted`	Required	62.0
`billingScheduleIds`	String[]	List of billing schedule IDs that's used to create the invoices. You can specify a maximum of 200 billing schedules.	Required if the `accountId` or `billing`	62.0

Name	Type	Description	Required or Optional	Available Version
			`TransactionId` property isn't specified.	
`billingTransactionId`	String	ID of the billing transaction record, which is the order ID, to create the invoices for.	Required if the `accountId` or `billingScheduleIds` property isn't specified.	63.0
`correlationId`	String	Property that's tagged against the published `InvoiceProcessedEvent` event, if specified.	Optional	62.0
`invoiceDate`	String	Stamping date of the invoice in ISO 8601 format.	Required	62.0
`targetDate`	String	Date in ISO 8601 format used to decide the billing periods that are included to create invoices.	Required	62.0

Invoice Preview Input

Input representation of the details of the billing transaction that the preview invoices are generated for.

JSON example

```
{
    "billingTransactionId": "801Z600000004LoIAI",
    "previewDate": "2024-12-04"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`billingTransactionId`	String	ID of the record to generate the preview invoices for.	Required	63.0
`previewDate`	String	The date on which the preview invoice is generated. For the first invoice, the preview date is the target date for generating the invoice. For the second invoice, the target date is calculated based on the preview	Optional	63.0

Name	Type	Description	Required or Optional	Available Version
		date and the minimum billing frequency of the transactions. The default value is the current date.		

Line Item Input

Input representation of the details of the line item for tax calculation.

JSON example

```
{
  "lineItems": [
    {
      "quantity": 1,
      "amount": 100,
      "taxCode": "TX0001",
      "productCode": "Y0001",
      "description": "New Product",
      "effectiveDate": "2022-03-09T10:30:41.000Z",
      "lineNumber": "5TVxx0000004C92GAE",
      "addresses": {
        "shipFrom": {
          "street": "123 Alaskan Way",
          "city": "Seattle",
          "state": "WA",
          "country": "US",
          "postalCode": "98101",
          "latitude": 45.12,
          "longitude": 45.12
        },
        "shipTo": {
          "street": "123 Auburn Blvd",
          "city": "Sacramento",
          "state": "CA",
          "country": "US",
          "postalCode": "95841",
          "latitude": 45.12,
          "longitude": 45.12
        },
        "soldTo": {
          "street": "123 Auburn Blvd",
          "city": "Sacramento",
          "state": "CA",
          "country": "US",
          "postalCode": "95841",
          "latitude": 45.12,
          "longitude": 45.12
        },
        "billTo": {
          "street": "123 Auburn Blvd",
```

```
          "city": "Sacramento",
          "state": "CA",
          "country": "US",
          "postalCode": "95841",
          "latitude": 45.12,
          "longitude": 45.12
        }
      }
    }
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
addresses	Addresses Input	Address types that specify different locations associated with the transaction of the line item.	Optional	62.0
amount	Double	Total amount for the line item.	Required	62.0
description	String	Description of the line item.	Optional	62.0
effectiveDate	String	Date that tax rules are applicable from. This property value overrides the global effective date.	Optional	62.0
lineNumber	String	Unique identifier for the line item.	Required	62.0
productCode	String	Product code of the line item according to the tax treatment rules of the tax engine.	Optional	62.0
quantity	Double	Quantity of the line item.	Required	62.0
taxCode	String	Tax code of the line item according to the tax treatment rules of the tax engine.	Optional	62.0

Resume Billing Input

Input representation of the details of the request to resume the billing operation for an account or a billing schedule group.

JSON example

```
{
    "referenceIds":
    [
        {
          "referenceId": "001DU000001o2UwYAI",
          "resumeDate": "2024-11-27"
        }
    ]

}
```

Properties

Name	Type	Description	Required or Optional	Available Version
referenceIds	Resume Billing Object Input[]	Input representation of the account or billing schedule group IDs to resume the billing operation for.	Required	63.0

Resume Billing Object Input

Input representation of the details such as the ID of the account or billing schedule group along with the effective date. These details are used to start the billing operation.

JSON example

```
{
    "referenceIds":
    [
        {
          "referenceId": "001DU000001o2UwYAI",
          "resumeDate": "2024-11-27"
        }
    ]

}
```

Properties

Name	Type	Description	Required or Optional	Available Version
referenceId	String	ID of the account or billing schedule group to resume the billing operation for.	Required	63.0
resumeDate	String	Date when the billing operation is resumed. If a date isn't specified, the default value is today's date. The billing operation starts immediately and any future suspension dates aren't applicable.	Required	63.0

Seller Details Input

Input representation of the seller details for tax calculation.

JSON example

```
{
  "code": "ADIDAS"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
code	String	Seller code as specified in the tax engine.	Required	62.0

Standalone Credit Memo Charge Input

Input representation of the details of the charge lines of a credit memo.

JSON example

```
"charges": [
  {
    "chargeAmount": 100,
    "productId": "01tR0000000njDiIAI"
  }
]
```

Properties

Name	Type	Description	Required or Optional	Available Version
addresses	Credit Memo Addresses Input	Details of the billing and shipping addresses.	Optional	62.0
chargeAmount	Double	Charge amount for the credit memo.	Required	62.0
description	String	Description of the created credit memo charge line.	Optional	62.0
endDate	String	End date of the credit memo charge line.	Optional	62.0
isTaxOnlyCredit	Boolean	Indicates whether the credit is for tax only (true) or not (false).	Optional	62.0
productId	String	ID of the product record that the credit memo is issued on.	Optional	62.0
productName	String	Name of the product that the credit memo is issued on.	Optional	62.0
startDate	String	Start date of the credit memo charge line.	Optional	62.0
taxEffectiveDate	String	Date from when the tax is applicable.	Optional	62.0
taxes	Standalone Credit Memo Tax Input[]	List of credit memo tax lines.	Optional	62.0
taxStrategy	String	Tax strategy to be applied to this credit memo charge line, child treatment lines, and tax lines. You can override the tax	Optional	62.0

Name	Type	Description	Required or Optional	Available Version
		strategy at the individual credit memo lines or tax lines level. Valid values are: • `Ignore`—Specifies that the creation of tax lines must be ignored. • `Calculate`—Specifies that tax must be calculated by using the API.		
`treatmentId`	String	ID of the tax treatment record that's used to calculate tax.	Optional	62.0

Standalone Credit Memo Input

Input representation of the details required to create a standalone credit memo.

JSON example

This example shows a sample request with the `Ignore` tax strategy.

```
{
  "billingAccountId": "001j000000WCFB800x",
  "type": "Posted",
  "description": "Standalone credit memo with ignored tax.",
  "currencyIsoCode": "USD",
  "taxStrategy": "Ignore",
  "charges": [
    {
      "chargeAmount": 100,
      "productId": "01tR0000000njDiIAI"
    }
  ]
}
```

This example shows a sample request with the `Calculate` tax strategy for a charge line.

```
{
  "billingAccountId": "001DU000001nhoPYAQ",
  "description": "Standalone Credit Memo",
  "type": "Posted",
  "currencyIsoCode": "USD",
  "taxStrategy": "Ignore",
  "charges": [
    {
      "productId": "01tR0000000njDiIAI",
      "chargeAmount": 10,
      "taxStrategy": "Calculate",
      "treatmentId": "1ttxx00000001VZAAY"
    }
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`billToContactId`	String	Contact related to the credit memo.	Optional	62.0
`billingAccountId`	String	ID of the account that the credit is issued to.	Required	62.0
`charges`	Standalone Credit Memo Charge Input[]	Charge lines of the credit memo. Requires at least one charge line.	Required	62.0
`currencyIsoCode`	String	ISO code currency of the new credit that's issued.	Optional	62.0
`description`	String	Description for the new credit that's issued.	Optional	62.0
`effectiveDate`	String	Effective date of the credit memo. If the value isn't specified, then it's null.	Optional	62.0
`externalReference`	String	ID of the external reference for the credit memo.	Optional	62.0
`externalReferenceDataSource`	String	Source of the external reference for the credit memo.	Optional	62.0
`taxEffectiveDate`	String	Effective date of the credit memo tax. If the value isn't specified, then it's null.	Optional	62.0
`taxStrategy`	String	Specifies how tax lines must be created for the standalone credit memos. Valid values are: • `Ignore`—Specifies that the creation of tax lines must be ignored. • `Calculate`—Specifies that tax must be calculated by using the API.	Required	62.0
`type`	String	Type of credit memo to be created. Valid value is `Posted`.	Optional	62.0

Standalone Credit Memo Tax Input

Input representation of the details of the tax request.

JSON example

```
[
  {
    "taxAmount": 200,
    "taxName": "Federal Tax",
```

```
    "taxRate": 1
  },
  {
    "taxAmount": 500,
    "taxName": "State Tax",
    "taxRate": 1
  }
]
```

Properties

Name	Type	Description	Required or Optional	Available Version
taxAmount	Double	Amount of tax to be applied.	Required	62.0
taxCode	String	Tax code to be used to create the tax line.	Optional	62.0
taxName	String	Name of tax to be applied.	Optional	62.0
taxRate	Double	Tax rate to be used to create the tax line.	Optional	62.0

Suspend Billing Input

Input representation of the details of the request to suspend the billing operation for an account or a billing schedule group.

JSON example

```
{
    "referenceIds":
    [
        {
            "referenceId": "001DU000001o2UwYAI",
            "suspendDate": "2024-11-27",
            "resumeDate": "2024-12-27"
        }
    ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
referenceIds	Suspend Billing Object Input[]	Input representation of the account or billing schedule group IDs to suspend the billing operation for.	Required	63.0

Suspend Billing Object Input

Input representation of the details such as the ID of the account or billing schedule group along with the effective dates. These details are used to suspend the billing operation.

JSON example

```
{
    "referenceIds":
    [
        {
            "referenceId": "001DU000001o2UwYAI",
            "suspendDate": "2024-11-27",
            "resumeDate": "2024-12-27"
        }
    ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
referenceId	String	ID of the account or billing schedule group to suspend the billing operation for.	Required	63.0
resumeDate	String	Date until when the account or billing schedule group is suspended for billing.	Required	63.0
suspendDate	String	Date when the account or billing schedule group is suspended for billing.	Required	63.0

Tax Calculation Input

Input representation of the details of the request to calculate tax.

JSON example

```
{
  "addresses": {
    "billTo": {
      "street": "123 Main Street",
      "city": "Bainbridge Island",
      "state": "WA",
      "postalCode": "98110",
      "country": "US"
    },
    "soldTo": {
      "street": "123 Main Street",
      "city": "Bainbridge Island",
      "state": "WA",
      "postalCode": "98110",
      "country": "US"
    },
    "shipFrom": {
      "street": "123 Alaskan Way",
      "city": "Seattle",
      "state": "WA",
      "country": "US",
```

```
    "postalCode": "98101"
  },
  "shipTo": {
    "street": "123 Main street",
    "city": "Bainbridge Island",
    "state": "WA",
    "postalCode": "98110",
    "country": "US"
  }
},
"currencyIsoCode": "USD",
"customerDetails": {
  "accountId": "001xx000003HYD5AAO"
},
"description": "Monthly invoice for account 001xx000003HYEhAAO",
"documentCode": "INV-003",
"effectiveDate": "2022-03-09T10:30:41.000Z",
"isCommit": true,
"lineItems": [
  {
    "quantity": 1,
    "amount": 100,
    "taxCode": "TX0001",
    "productCode": "Y0001",
    "description": "New Product",
    "effectiveDate": "2022-03-09T10:30:41.000Z",
    "lineNumber": "5TVxx0000004C92GAE",
    "addresses": {
      "shipFrom": {
        "street": "123 Alaskan Way",
        "city": "Seattle",
        "state": "WA",
        "country": "US",
        "postalCode": "98101",
        "latitude": 45.12,
        "longitude": 45.12
      },
      "shipTo": {
        "street": "123 Auburn Blvd",
        "city": "Sacramento",
        "state": "CA",
        "country": "US",
        "postalCode": "95841",
        "latitude": 45.12,
        "longitude": 45.12
      },
      "soldTo": {
        "street": "123 Auburn Blvd",
        "city": "Sacramento",
        "state": "CA",
        "country": "US",
        "postalCode": "95841",
        "latitude": 45.12,
        "longitude": 45.12
```

```
        },
        "billTo": {
          "street": "123 Auburn Blvd",
          "city": "Sacramento",
          "state": "CA",
          "country": "US",
          "postalCode": "95841",
          "latitude": 45.12,
          "longitude": 45.12
        }
      }
    }
  ],
  "referenceEntityId": "001xx000003HYEhAAO",
  "taxEngineId": "4wAxx0000000g8v",
  "taxType": "Actual",
  "taxTransactionType": "Debit",
  "transactionDate": "2022-03-09T10:30:41.000Z"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
`addresses`	Addresses	Address details for tax calculation.	Optional	62.0
`currencyIsoCode`	String	Currency ISO code that's used for tax calculation.	Optional	62.0
`customerDetails`	Customer Details	Customer details for determining the applicable tax.	Optional	62.0
`description`	String	Description of the tax transaction.	Optional	62.0
`documentCode`	String	Unique identifier for the tax document. If the `documentCode` property isn't specified, the tax engine auto-generates it.	Optional	62.0
`effectiveDate`	String	Date when the tax rules are applied. If a tax rate changes on a specific date, the `effectiveDate` property ensures the correct rate is applied based on the transaction's timing.	Optional	62.0
`isCommit`	Boolean	Indicates whether to submit the transaction to the tax engine for reporting (`true`) or not (`false`). This property value is `true` for invoices and credit memos with the `Posted` status.	Optional	62.0

Name	Type	Description	Required or Optional	Available Version
`lineItems`	Line Item Input []	Details of the line items for calculating the applicable tax.	Required	62.0
`reference DocumentCode`	String	Reference document code. For subsequent transactions such as a credit tax, this property value specifies the original document code.	Optional	62.0
`reference EntityId`	String	ID of the related quote, invoice, and other transaction documents.	Optional	62.0
`sellerDetails`	Seller Details Input	Seller details for tax calculation.	Optional	62.0
`taxEngineId`	String	ID of the tax engine that's used to calculate tax.	Required	62.0
`tax TransactionType`	String	Type of the tax transaction. Valid values are: • `Debit`—Increases tax liability for the seller, requiring the seller to pay tax on the transaction. • `Credit`—Decreases tax liability for the seller, resulting in a tax refund for the seller. The default value is `Debit`.	Optional	62.0
`taxType`	String	Type of the tax. Valid values are: • `Actual`—Exact tax amount that's calculated based on actual sales. • `Estimated`— Estimated tax amount, which is adjusted later to match actual tax calculations. For draft invoices and quote records, the tax type is marked as `Estimated`. After draft invoices are posted and the status changes to `Posted`, the tax type is updated to `Actual`. Similarly, when a quote record is finalized and converted into an actual order, the tax type is also updated to `Actual`.	Required	62.0
`transactionDate`	String	Date of the transaction that appears on the invoice, order, and other transaction documents.	Required	62.0

Void Posted Invoice Input

Input representation of the details of the invoice to be voided.

Properties

Name	Type	Description	Required or Optional	Available Version
`invoiceId`	String	ID of the posted invoice to be voided.	Required	62.0

Response Bodies

Learn more about the available response bodies of Billing APIs.

Addresses
Output representation of the details of the addresses that are used for calculating tax.

Address
Output representation of the location code associated with an address.

Batch Invoice Document Generation
Output representation of the request to generate or regenerate the PDF documents for the invoices that are in the `Draft` or `Posted` status.

Error Response for Batch Invoice Document Generation
Output representation of the error details associated with the Batch Invoice Document Generation API.

Batch Invoice Scheduler
Output representation of the details of an invoice scheduler.

Billing Batch Scheduler
Output representation of the details of a created invoice scheduler.

Billing Schedule Recovery List
Output representation of the recovered details of the billing schedules and associated invoice.

Billing Schedule Recovery
Output representation of the details of the recovered billing schedules.

Context-Aware Billing Schedule
Output representation of the context-aware billing schedule.

Context-Aware Billing Schedule Error
Output representation of the error response related to the generation of the billing schedule.

Convert Negative Invoice Lines
Output representation of the details of the created memo along with the status of the request.

Credit Memo Apply
Output representation of the list of applied credit memo results.

Credit Memo Apply List
Output representation of the list of applied credit memo results.

Credit Memo Unapply

Output representation of the details of the credit memo invoice application record with the status of the request.

Credit Memo Line Applied

Output representation of the list of applied credit memo line results.

Credit Memo Line Applied Response

Output representation of the list of applied credit memo line results.

Credit Memo Line Unapplied

Output representation of the details of the credit memo line invoice line record with the status of the request.

Invoice Batch Draft To Posted

Output representation of the batch update details of the invoices from `Draft` to `Posted` status.

Invoice Batch Run Recovery

Output representation of the details of the invoice batch run recovery record.

Invoice Ingestion

Output representation of the details of the generated invoices.

Invoice Ingestion Details

Output representation of the details of a generated invoice.

Invoice Ingestion Output Error

Output representation of the details of an invoice generation error.

Invoice Line Preview

Output representation of the invoice line preview result.

Invoice Preview

Output representation of the invoice preview result.

Invoice Preview Result

Output representation of the list of preview invoices that are generated for the billing transaction.

Invoice Recovery

Output representation of the details of the recovered invoice and billing schedules.

Line Item

Output representation of the details of the line item.

Reference Line Error

Output representation of the details of the line level errors.

Revenue Async Line Level

Output representation of the result of the API request for the async line level operations.

Revenue Async Response

Output representation of the result of the API request with the request identifier.

Suspend Resume Billing

Output representation of the list of accounts and billing schedule groups, which are suspended or resumed for billing operations.

Suspend Resume Billing Object

Output representation of the details of accounts and billing schedule groups, which are suspended or resumed for billing operations, along with the status of the API request.

Tax Amount Details

Output representation of the details of the tax amount.

Tax Calculation
Output representation of the details of the calculated tax.

Tax Details
Output representation of the tax details for each line item.

Tax Engine Log
Output representation of the logs that the tax engine generates.

Tax Imposition
Output representation of the details of the imposed tax.

Tax Jurisdiction
Output representation of the details of the tax jurisdiction for the tax line.

Addresses

Output representation of the details of the addresses that are used for calculating tax.

JSON example

```
{
  "addresses": {
    "shipFrom": {
      "locationCode": "67890"
    },
    "shipTo": {
      "locationCode": "12345"
    },
    "soldTo": {
      "locationCode": "12345"
    }
  }
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
shipFrom	Address on page 1606	Address that the item is shipped from.	Big, 62.0	62.0
shipTo	Address on page 1606	Address that the item is shipped to.	Big, 62.0	62.0
soldTo	Address on page 1606	Address that the item is sold to.	Big, 62.0	62.0

Address

Output representation of the location code associated with an address.

JSON example

```
{
    "locationCode": "123456789"
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
locationCode	String	Unique location code of the address.	Big, 62.0	62.0

Batch Invoice Document Generation

Output representation of the request to generate or regenerate the PDF documents for the invoices that are in the `Draft` or `Posted` status.

JSON example

```
{
  "errors": [
    {
      "errorCode": "API_DISABLED_FOR_ORG",
      "errorMessage": "Document Generation is not enabled for this org!"
    },
    {
      "errorCode": "INVALID_API_INPUT",
      "errorMessage": "Invalid Invoice Batch Run Id"
    }
  ],
  "requestIdentifier": "5IRDU000000009i4AA",
  "success": false
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
errors	Batch Invoice Document Generation Error []	Details of the error if the operation fails.	Big, 63.0	63.0
requestIdentifier	String	Unique ID that's associated with the specific error, and is used for tracking and referencing the request.	Big, 63.0	63.0
success	Boolean	Indicates whether the operation is successful (`true`) or not (`false`).	Big, 63.0	63.0

Error Response for Batch Invoice Document Generation

Output representation of the error details associated with the Batch Invoice Document Generation API.

JSON example

```
  "errors": {
    "errorCode": "API_DISABLED_FOR_ORG",
    "message": "Document Generation is not enabled for this org!"
  }
```

Property Name	Type	Description	Filter Group and Version	Available Version
errorCode	String	Code for the resultant error.	Big, 63.0	63.0
errorMessage	String	Error message for the resultant error.	Big, 63.0	63.0

Batch Invoice Scheduler

Output representation of the details of an invoice scheduler.

JSON example

```
{
    "billingBatchScheduler": {
        "id": "5BSxx0000004TwGGAU"
    }
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
billingBatchScheduler	Billing Batch Scheduler[]	Details of the created invoice scheduler.	Big, 62.0	62.0

Billing Batch Scheduler

Output representation of the details of a created invoice scheduler.

JSON example

```
    "billingBatchScheduler": {
        "id": "5BSxx0000004TwGGAU"
    }
```

Property Name	Type	Description	Filter Group and Version	Available Version
id	String	ID of the invoice scheduler.	Big, 62.0	62.0

Billing Schedule Recovery List

Output representation of the recovered details of the billing schedules and associated invoice.

JSON example

```
{
  "recoveryResults": [
    {
      "billingSchedules": [
        {
          "billingScheduleId": "44bDU00000000XX",
```

```
          "billingScheduleStatus": "ReadyForInvoicing"
        }
      ],
      "invoiceErrors": [],
      "invoiceId": null,
      "success": true
    }
  ]
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
recoveryResults	Invoice Recovery[]	Details of the recovered invoice and billing schedules.	Big, 62.0	62.0

Billing Schedule Recovery

Output representation of the details of the recovered billing schedules.

JSON example

```
    "billingSchedules": [
        {
          "billingScheduleId": "44bDU00000000XX",
          "billingScheduleStatus": "ReadyForInvoicing"
        }
    ]
```

Property Name	Type	Description	Filter Group and Version	Available Version
billingScheduleId	String	ID of the billing schedule.	Big, 62.0	62.0
billingScheduleStatus	String	Flag that indicates the billing schedule status.	Big, 62.0	62.0

Context-Aware Billing Schedule

Output representation of the context-aware billing schedule.

JSON example

This request shows a sample success response.

```
{
    "errors": null,
    "requestIdentifier": "16Pxx0000004CaS",
   "statusURL": "/services/data/v63.0/sobjects/AsyncOperationTracker/16Pxx0000004CaSEAU",
```

```
    "success": true
}
```

This request shows a sample error response.

```
{
    "errors": [
        {
            "errorCode": "REQUIRED_FIELD_MISSING",
            "errorMessage": "Required fields are missing: billToContact",
            "referenceId": "802xx000001nmb5"
        }
    ],
    "requestIdentifier": "16Pxx0000004CYq",
   "statusURL": "/services/data/v63.0/sobjects/AsyncOperationTracker/16Pxx0000004CYqEAM",

    "success": false
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
errors	Context Aware Billing Schedule Error[]	Error response if the generation of the billing schedule fails.	Big, 62.0	62.0
requestIdentifier	String	Unique request identifier that you can use to poll the asynchronous request.	Big, 62.0	62.0
statusURL	String	Status URL to track the operation.	Big, 62.0	62.0
success	Boolean	Indicates whether the processing of the billing schedule is successful (true) or not (false).	Big, 62.0	62.0

Context-Aware Billing Schedule Error

Output representation of the error response related to the generation of the billing schedule.

```
    "errors": [
        {
            "errorCode": "REQUIRED_FIELD_MISSING",
            "errorMessage": "Required fields are missing: billToContact",
            "referenceId": "802xx000001nmb5"
        }
    ]
```

Property Name	Type	Description	Filter Group and Version	Available Version
errorCode	String	Code for the resultant error.	Big, 62.0	62.0
errorMessage	String	Error message for the resultant error.	Big, 62.0	62.0

Property Name	Type	Description	Filter Group and Version	Available Version
referenceId	String	Reference ID of the source item that resulted in the error.	Big, 62.0	62.0

Convert Negative Invoice Lines

Output representation of the details of the created memo along with the status of the request.

JSON example

```
{
"id": "50gxx000000g0WwAAI",
"success": true,
"errors": []
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
errors	Error Response	List of errors encountered during the processing of the API request.	Big, 62.0	62.0
id	String	ID of the credit memo that's created after the conversion of the negative invoice lines.	Small, 62.0	62.0
success	Boolean	Indicates whether the API request is successful (true) or not (false).	Small, 62.0	62.0

Credit Memo Apply

Output representation of the list of applied credit memo results.

JSON example

```
"applyCreditResults" : [ {
  "appliedToId" : "3ttxx000000003FAAQ",
  "errors" : null,
  "id" : "4sFxx00000002ppEAA",
  "success" : true
}, {
  "appliedToId" : "3ttxx0000000001AAA",
  "errors" : null,
  "id" : "4sFxx00000002pqEAA",
  "success" : true
} ]
```

Property Name	Type	Description	Filter Group and Version	Available Version
appliedToId	String	ID of the invoice record that the credit is applied to.	Big, 62.0	62.0

Property Name	Type	Description	Filter Group and Version	Available Version
errors	Error Response	List of errors encountered during the processing of the API request.	Big, 62.0	62.0
id	String	ID of the credit memo invoice application.	Big, 62.0	62.0
success	Boolean	Indicates whether the credit memo is successfully applied (true) or not (false).	Big, 62.0	62.0

Credit Memo Apply List

Output representation of the list of applied credit memo results.

JSON example

```
{
  "applyCreditResults" : [ {
    "appliedToId" : "3ttxx000000003FAAQ",
    "errors" : null,
    "id" : "4sFxx00000002ppEAA",
    "success" : true
  }, {
    "appliedToId" : "3ttxx0000000001AAA",
    "errors" : null,
    "id" : "4sFxx00000002pqEAA",
    "success" : true
  } ]
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
applyCreditResults	Credit Memo Apply[]	Output list of the applied credit memo results.	Big, 62.0	62.0

Credit Memo Unapply

Output representation of the details of the credit memo invoice application record with the status of the request.

JSON example

```
{
    "errors": [],
    "id": "4sFxx00000002ppEAA",
    "success": true
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
errors	Error Response	List of errors encountered during the processing of the API request.	Big, 62.0	62.0
id	String	ID of the credit memo invoice application record.	Small, 62.0	62.0
success	Boolean	Indicates whether the credit memo is successfully unapplied (true) or not (false).	Small, 62.0	62.0

Credit Memo Line Applied

Output representation of the list of applied credit memo line results.

JSON example

```
{
 "appliedCreditResponses": [
  {
   "creditMemoLineInvoiceLineId": "4sGSG0000002pMb2AI",
   "errors": null,
   "invoiceLineId": "5TVSG0000003CuH4AU",
   "success": true
  }
 ]
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
appliedCreditResponses	Credit Memo Line Applied Response[]	Output list for the applied credit memo line results.	Big, 62.0	62.0

Credit Memo Line Applied Response

Output representation of the list of applied credit memo line results.

JSON example

```
"appliedCreditResponses": [
 {
  "creditMemoLineInvoiceLineId": "4sGSG0000002pMb2AI",
  "errors": null,
  "invoiceLineId": "5TVSG0000003CuH4AU",
  "success": true
 }
]
```

Property Name	Type	Description	Filter Group and Version	Available Version
`creditMemoLineInvoiceLineId`	String	ID of the credit memo line invoice line ID.	Big, 62.0	62.0
`errors`	Error Response	List of errors encountered during the processing of the API request.	Big, 62.0	62.0
`invoiceLineId`	String	ID of the invoice line record that the credit is applied to.	Big, 62.0	62.0
`success`	Boolean	Indicates whether the credit memo line is successfully applied (`true`) or not (`false`).	Big, 62.0	62.0

Credit Memo Line Unapplied

Output representation of the details of the credit memo line invoice line record with the status of the request.

JSON example

```
 {
 "creditMemoLineInvoiceLineId": "4sGSG0000002pOD2AY",
 "errors": [],
 "success": true
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`creditMemoLineInvoiceLineId`	String	ID of the credit memo line invoice line record.	Small, 62.0	62.0
`errors`	Error Response	List of errors encountered during the processing of the API request.	Big, 62.0	62.0
`success`	Boolean	Indicates whether the credit memo line is successfully unapplied (`true`) or not (`false`).	Small, 62.0	62.0

Invoice Batch Draft To Posted

Output representation of the batch update details of the invoices from `Draft` to `Posted` status.

JSON example

```
{
  "invoiceBatchDraftToPostedId": "4sFDU00000000652AA",
  "success": true
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`invoiceBatchDraftToPostedId`	String	ID of the invoice batch draft to posted run record that's created to track the batch process of posting the draft invoices that are associated with the parent invoice batch run record.	Small, 62.0	62.0
`success`	Boolean	Indicates whether the API request is successful (`true`) or not (`false`).	Small, 62.0	62.0

Invoice Batch Run Recovery

Output representation of the details of the invoice batch run recovery record.

JSON example

```
{
  "invoiceBatchRunRecoveryId": ["0wBxx0000004TtwGAU"]
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`invoiceBatchRunRecoveryId`	String	ID of the invoice batch run recovery record for the specified invoice batch run. This record represents the background recovery process.	Small, 62.0	62.0

Invoice Ingestion

Output representation of the details of the generated invoices.

JSON example

```
{
  "invoices": [
    {
      "errors": null,
      "invoiceId": "3ttxx0000000GRNAA2",
      "requestIdentifier": "16e80fbc-e7b3-4462-9439-745647fcf0a8",
      "statusURL":
"/services/data/v63.0/sobjects/AsyncOperationTracker/16Pxx0000004Gz2EAE",
      "success": true
    }
  ]
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
invoices	Invoice Ingestion []	Result that contains the details for each generated invoice.	Big, 63.0	63.0

Invoice Ingestion Details

Output representation of the details of a generated invoice.

JSON example

```
{
  "invoices": [
    {
      "errors": null,
      "invoiceId": "3ttxx0000000GRNAA2",
      "requestIdentifier": "16e80fbc-e7b3-4462-9439-745647fcf0a8",
      "statusURL":
"/services/data/v63.0/sobjects/AsyncOperationTracker/16Pxx0000004Gz2EAE",
      "success": true
    }
  ]
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
errors	Invoice Ingestion Output Error[]	Details of the errors if the request was unsuccessful.	Small, 63.0	63.0
invoiceId	String	ID of the Invoice record that's created in Salesforce.	Small, 63.0	63.0
request Identifier	String	Unique request identifier that can be used to poll the async request.	Small, 63.0	63.0
statusURL	String	Status URL for tracking the estimated tax callout operation.	Big, 63.0	63.0
success	Boolean	Indicates whether the API request was successful (true) or not (false).	Small, 63.0	63.0

Invoice Ingestion Output Error

Output representation of the details of an invoice generation error.

Property Name	Type	Description	Filter Group and Version	Available Version
errorCode	String	Code that indicates the type of error.	Small, 63.0	63.0

Property Name	Type	Description	Filter Group and Version	Available Version
message	String	Message that states the reason for error, if any.	Small, 63.0	63.0
referenceId	String	Reference ID that maps to the subrequest's response and can be used to reference the response in later subrequests.	Small, 63.0	63.0

Invoice Line Preview

Output representation of the invoice line preview result.

JSON example

```
"invoiceLineDetailList": [
      {
        "billingFrequency": "OneTime",
        "chargeAmount": "1990.00",
        "endDate": "2024-07-18",
        "lineAmount": "2189.00",
        "productName": "Laptop",
        "quantity": "10.0",
        "startDate": "2024-07-18",
        "taxAmount": "199.0",
        "unitPrice": "199.0"
      }
    ]
```

Property Name	Type	Description	Filter Group and Version	Available Version
billing Frequency	String	The frequency at which the invoice line is created for the billing transaction.	Big, 63.0	63.0
endDate	String	The end date of the billing for the service for invoice lines made from a time-based service.	Big, 63.0	63.0
lineAmount	String	The amount of the invoice line.	Big, 63.0	63.0
productName	String	The name of the product that was charged or ordered to create the invoice line.	Big, 63.0	63.0
quantity	String	The number of units of the order product that created the invoice line.	Big, 63.0	63.0
startDate	String	The first date of the billing for the service for invoice lines made from a time-based service.	Big, 63.0	63.0
taxAmount	String	The tax amount for the invoice line.	Big, 63.0	63.0

Property Name	Type	Description	Filter Group and Version	Available Version
unitPrice	String	The price for one unit of the item on the invoice line.	Big, 63.0	63.0

Invoice Preview

Output representation of the invoice preview result.

JSON example

```
"invoiceDetailList": [
  {
    "accountId": "001Z6000005aQj8",
    "currencyIsoCode": "USD",
    "dueDate": "2025-01-03",
    "invoiceDate": "2024-12-04",
    "invoiceLineDetailList": [
      {
        "billingFrequency": "OneTime",
        "chargeAmount": "1990.00",
        "endDate": "2024-07-18",
        "lineAmount": "2189.00",
        "productName": "Laptop",
        "quantity": "10.0",
        "startDate": "2024-07-18",
        "taxAmount": "199.0",
        "unitPrice": "199.0"
      }
    ],
    "totalAmount": "1990.00",
    "totalAmountWithTax": "2189.00",
    "totalTaxAmount": "199.0"
  }
]
```

Property Name	Type	Description	Filter Group and Version	Available Version
accountId	String	The account ID of the billing transaction.	Big, 63.0	63.0
currencyCode	String	The currency code for the amount fields.	Big, 63.0	63.0
dueDate	String	The date by when the customer must pay an invoice that's generated for the billing transaction.	Big, 63.0	63.0
invoiceDate	String	The date when an invoice that's generated for the billing transaction is posted.	Big, 63.0	63.0
invoiceLines	Invoice Line Preview[]	The details of the invoice lines related to the invoices that are generated for the billing transaction.	Big, 63.0	63.0

Property Name	Type	Description	Filter Group and Version	Available Version
totalAmount	String	The sum of the total amount on the invoice lines of an invoice that's generated for the billing transaction.	Big, 63.0	63.0
totalAmountWithTax	String	The sum of the total amount including tax on the invoice lines of an invoice that's generated for the billing transaction.	Big, 63.0	63.0
totalTaxAmount	String	The sum of the tax amount on the invoice lines of an invoice that's generated for the billing transaction.	Big, 63.0	63.0

Invoice Preview Result

Output representation of the list of preview invoices that are generated for the billing transaction.

JSON example

```
{
  "invoiceDetailList": [
    {
      "accountId": "001Z6000005aQj8",
      "currencyIsoCode": "USD",
      "dueDate": "2025-01-03",
      "invoiceDate": "2024-12-04",
      "invoiceLineDetailList": [
        {
          "billingFrequency": "OneTime",
          "chargeAmount": "1990.00",
          "endDate": "2024-07-18",
          "lineAmount": "2189.00",
          "productName": "Laptop",
          "quantity": "10.0",
          "startDate": "2024-07-18",
          "taxAmount": "199.0",
          "unitPrice": "199.0"
        }
      ],
      "totalAmount": "1990.00",
      "totalAmountWithTax": "2189.00",
      "totalTaxAmount": "199.0"
    }
  ]
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
invoiceDetailList	Invoice Preview[]	The details of the invoices that are generated for the billing transaction.	Big, 63.0	63.0

Invoice Recovery

Output representation of the details of the recovered invoice and billing schedules.

JSON example

```
{
  "recoveryResults": [
    {
      "billingSchedules": [
        {
          "billingScheduleId": "44bDU00000000XX",
          "billingScheduleStatus": "ReadyForInvoicing"
        }
      ],
      "invoiceErrors": [],
      "invoiceId": null,
      "success": true
    }
  ]
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`billingSchedules`	Billing Schedule Recovery[]	Billing schedules associated with this invoice.	Big, 62.0	62.0
`invoiceErrors`	Error Response	List of errors encountered during the invoice recovery.	Big, 62.0	62.0
`invoiceId`	String	ID of the recovered invoice.	Big, 62.0	62.0
`invoiceStatus`	String	Flag that indicates the invoice status.	Big, 62.0	62.0
`success`	Boolean	Indicates whether the overall transaction was successful or not (`true`) or not (`false`).	Big, 62.0	62.0

Line Item

Output representation of the details of the line item.

JSON example

```
{
  "lineItems": [
    {
      "addresses": {
        "shipFrom": {
          "locationCode": "67890"
        },
        "shipTo": {
          "locationCode": "12345"
        },
```

```
        "soldTo": {
          "locationCode": "12345"
        }
      },
      "amountDetails": {
        "exemptAmount": 0,
        "taxAmount": 12.5,
        "totalAmount": 100,
        "totalAmountWithTax": 112.5
      },
      "effectiveDate": "2022-03-09T10:55:38.416Z",
      "lineNumber": "001xx000003HYEiAAO",
      "productCode": "Y0001",
      "quantity": 1,
      "taxCode": "TX0001",
      "taxes": [
        {
          "exemptAmount": 0,
          "exemptReason": "NoExemption",
          "imposition": {
            "type": "General"
          },
          "jurisdiction": {
            "country": "US",
            "id": "63000",
            "level": "CIT",
            "name": "SEATTLE",
            "region": "WA",
            "stateAssignedNo": "1726"
          },
          "rate": 12.5,
          "tax": 12.5,
          "taxId": "11000378132466",
          "taxableAmount": 100
        }
      ]
    }
  ]
}
```


Property Name	Type	Description	Filter Group and Version	Available Version
`addresses`	Addresses	Different types of addresses that are used for the transaction of the line item.	Big, 62.0	62.0
`amountDetails`	Tax Amount Details	Details of the transaction amount and the taxes applicable for the line item.	Big, 62.0	62.0
`effectiveDate`	String	Date when the tax rules are applied on the line item.	Big, 62.0	62.0
`lineNumber`	String	Unique identifier of the line item.	Big, 62.0	62.0

Property Name	Type	Description	Filter Group and Version	Available Version
productCode	String	Product code of the line item as defined by the tax engine.	Big, 62.0	62.0
quantity	Double	Quantity of the line item.	Big, 62.0	62.0
taxCode	String	Tax code of the line item according to the tax engine.	Big, 62.0	62.0
taxes	Tax Details []	Tax details of the line item.	Big, 62.0	62.0

Reference Line Error

Output representation of the details of the line level errors.

JSON example

If the API request fails, the `referenceLineErrorResults` property contains a list of errors grouped by the invoice line IDs.

```
[
  {
    "referenceLineId": "5TV9A000007x2gz",
    "errors": [
      {
        "errorCode": "INVALID_INPUT",
        "message": "Invalid invoice line id"
      }
    ]
  }
]
```

Property Name	Type	Description	Filter Group and Version	Available Version
errors	Error Response	List of errors with error code and error message for the specified invoice line ID.	Big, 62.0	62.0
referenceLineId	String	ID of the invoice line specified in the API request that has an issue, causing the API request to fail.	Small, 62.0	62.0

Revenue Async Line Level

Output representation of the result of the API request for the async line level operations.

JSON example

```
  {
  "success": true,
  "requestIdentifier": "237e9877-e79b-12d4-a765-321741963000",
```

```
  "errors":[]
}
```

If the API request fails, the `referenceLineErrorResults` property contains a list of errors grouped by the invoice line IDs.

```
[
  {
    "referenceLineId": "5TV9A000007x2gz",
    "errors": [
      {
        "errorCode": "INVALID_INPUT",
        "message": "Invalid invoice line id"
      }
    ]
  }
]
```

Property Name	Type	Description	Filter Group and Version	Available Version
`errors`	Error Response	Details of errors, if any.	Big, 62.0	62.0
`referenceLineErrorResults`	Reference Line Error[]	List of errors grouped by the invoice line IDs if the API request fails.	Big, 62.0	62.0
`referenceLineType`	String	Reference type for the reference line entity in the `referenceLineErrorResults` property.	Big, 62.0	62.0
`requestIdentifier`	String	Unique identifier of the request.	Big, 62.0	62.0
`statusURL`	String	URL to track the status of the operation.	Big, 62.0	62.0
`success`	Boolean	Indicates whether the API request is successful (`true`) or not (`false`).	Big, 62.0	62.0

Revenue Async Response

Output representation of the result of the API request with the request identifier.

JSON example

```
  {
    "errors": null,
    "requestIdentifier": "ae6f23bc-f056-44b7-aa4d-c7f6fc5e0cf4",
    "success": true
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
errors	Error Response	Details of errors, if any.	Big, 62.0	62.0
requestIdentifier	String	Unique identifier of the request.	Big, 62.0	62.0
success	Boolean	Indicates whether the API request is successful (true) or not (false).	Big, 62.0	62.0

Suspend Resume Billing

Output representation of the list of accounts and billing schedule groups, which are suspended or resumed for billing operations.

JSON example

```
{
    "result":{
        {
            "referenceId": "001DU000001o2UwYAI",
            "success": true,
            "errorcode": null,
            "errorMessage":null
        },
        {
            "referenceId": "001DU000001o2UuYAI",
            "success": false,
            "errorcode": "INVALID_API_INPUT",
            "errorMessage":"Billing is already suspended for 9Wsxx000000006TCAQ."
        }
    }
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
result	Suspend Resume Billing Object[]	Details of the accounts or billing schedule groups that the suspend or resume billing request is initiated for.	Big, 63.0	63.0

Suspend Resume Billing Object

Output representation of the details of accounts and billing schedule groups, which are suspended or resumed for billing operations, along with the status of the API request.

JSON example

```
{
    "result":{
        {
        "referenceId": "1",
        "isSuccess": true,
```

```
        "errorcode": null,
        "errorMessage":null
        }
    }
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
errorCode	String	Code indicating the type of error.	Big, 63.0	63.0
errorMessage	String	Message stating the reason for the error, if any.	Big, 63.0	63.0
isSuccess	Boolean	Indicates whether the API request is successful (true) or not (false).	Big, 63.0	63.0
referenceId	String	ID of the account or billing schedule group that the suspend or resume billing request is initiated for.	Big, 63.0	63.0

Tax Amount Details

Output representation of the details of the tax amount.

JSON example

```
{
    "exemptAmount": 0.0,
    "taxAmount": 12.5,
    "totalAmount": 100.0,
    "totalAmountWithTax": 112.5
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
exemptAmount	Double	Amount exempted from taxation.	Big, 62.0	62.0
taxAmount	Double	Tax amount applicable to the transaction.	Big, 62.0	62.0
totalAmount	Double	Total amount without tax.	Big, 62.0	62.0
totalAmount WithTax	Double	Total amount with tax. The totalAmountWithTax property value is the sum of the taxAmount and totalAmount property values.	Big, 62.0	62.0

Tax Calculation

Output representation of the details of the calculated tax.

JSON example


```
{
    "referenceEntityId": "001xx000003HYEhAAO",
    "status": "Committed",
    "taxEngineLogs": [
        {
            "createdDate": "2022-03-09T10:55:38.000Z",
            "id": "3l1xx00000000PpAAI",
            "resultCode": "Success"
        }
    ],
    "taxTransactionType": "Debit",
    "taxType": "Actual",
    "transactionDate": "2022-03-09T10:30:41.000Z",
    "addresses": {
        "shipFrom": {
            "locationCode": "67890"
        },
        "shipTo": {
            "locationCode": "23456"
        },
        "soldTo": {
            "locationCode": "23456"
        }
    },
    "amountDetails": {
        "exemptAmount": 0.0,
        "taxAmount": 12.5,
        "totalAmount": 100.0,
        "totalAmountWithTax": 112.5
    },
    "currencyIsoCode": "USD",
    "description": "Monthly invoice for account 001xx000003HYEhAAO",
    "documentCode": "INV-003",
    "effectiveDate": "2022-03-09T10:55:38.410Z",
    "lineItems": [
        {
            "addresses": {
                "shipFrom": {
                    "locationCode": "67890"
                },
                "shipTo": {
                    "locationCode": "12345"
                },
                "soldTo": {
                    "locationCode": "12345"
                }
            },
            "amountDetails": {
                "exemptAmount": 0.0,
                "taxAmount": 12.5,
                "totalAmount": 100.0,
                "totalAmountWithTax": 112.5
```

```
            },
            "effectiveDate": "2022-03-09T10:55:38.416Z",
            "lineNumber": "001xx000003HYEiAAO",
            "productCode": "Y0001",
            "quantity": 1.0,
            "taxCode": "TX0001",
            "taxes": [
                {
                    "exemptAmount": 0.0,
                    "exemptReason": "NoExemption",
                    "imposition": {
                        "type": "General"
                    },
                    "jurisdiction": {
                        "country": "US",
                        "id": "63000",
                        "level": "CIT",
                        "name": "SEATTLE",
                        "region": "WA",
                        "stateAssignedNo": "1726"
                    },
                    "rate": 12.5,
                    "tax": 12.5,
                    "taxId": "11000378132466",
                    "taxableAmount": 100.0
                }
            ]
        }
    ]
}
```


Property Name	Type	Description	Filter Group and Version	Available Version
adapterError	Error Response	Details of the adapter error.	Big, 62.0	62.0
addresses	Addresses	Addresses that are used for calculating tax.	Big, 62.0	62.0
amountDetails	Tax Amount Details	Details of the transaction amount and taxes.	Big, 62.0	62.0
currencyIsoCode	String	Currency ISO code that's used for tax calculation.	Big, 62.0	62.0
description	String	Description of the transaction status.	Big, 62.0	62.0
documentCode	String	Unique identifier for the tax document. If the `documentCode` property isn't specified, the tax engine auto-generates it.	Big, 62.0	62.0
effectiveDate	String	Date when the tax rules are applied.	Big, 62.0	62.0
lineItems	Line Item []	Line items that the taxes are calculated for.	Big, 62.0	62.0
reference DocumentCode	String	Reference document code. For subsequent transactions such as a credit tax, the	Big, 62.0	62.0

Property Name	Type	Description	Filter Group and Version	Available Version
		`referenceDocumentCode` property value refers to the original document code.		
`referenceEntityId`	String	ID of the related quote, invoice, or other transaction documents.	Big, 62.0	62.0
`status`	String	Commit status of the tax transaction. Valid values are: • `Committed`—Committed transaction, which is marked for tax reporting. • `Uncommitted`— Uncommitted transaction, which isn't marked for tax reporting.	Big, 62.0	62.0
`taxEngineLogs`	Tax Engine Log []	Records that the tax engine generates while calculating taxes for the transaction.	Big, 62.0	62.0
`taxTransactionId`	String	Unique ID for the tax transaction.	Big, 62.0	62.0
`transactionDate`	String	Date of the transaction that appears on the invoice, order, and other transaction documents.	Big, 62.0	62.0
`taxTransactionType`	String	Tax transaction type. Valid values are: • `Debit`—Increases tax liability for the user, requiring the user to pay tax on the transaction. • `Credit`—Decreases tax liability for the user, resulting in a tax refund for the user.	Big, 62.0	62.0
`taxType`	String	Type of calculated tax. Valid values are: • `Actual`—Exact tax amount that's calculated based on the actual sales. • `Estimated`— Estimated tax amount, which is adjusted later to match the actual tax calculations.	Big, 62.0	62.0

Tax Details

Output representation of the tax details for each line item.

JSON example

```
{
  "taxes": [
```

```
    {
      "exemptAmount": 0,
      "exemptReason": "NoExemption",
      "imposition": {
        "type": "General"
      },
      "jurisdiction": {
        "country": "US",
        "id": "63000",
        "level": "CIT",
        "name": "SEATTLE",
        "region": "WA",
        "stateAssignedNo": "1726"
      },
      "rate": 12.5,
      "tax": 12.5,
      "taxId": "11000378132466",
      "taxableAmount": 100
    }
  ]
}
```


Property Name	Type	Description	Filter Group and Version	Available Version
exemptAmount	Double	Amount exempted from taxation for the line item.	Big, 62.0	62.0
exemptReason	String	Reason for the tax exemption.	Big, 62.0	62.0
imposition	Tax Imposition	Tax imposition details applicable to the line item.	Big, 62.0	62.0
jurisdiction	Tax Jurisdiction	Details of the tax jurisdiction.	Big, 62.0	62.0
rate	Double	Tax rate applied to the taxable amount to calculate the tax.	Big, 62.0	62.0
tax	Double	Actual amount of tax that's applicable to the line item.	Big, 62.0	62.0
taxId	String	Unique identifier such as a code or a number that's assigned to a specific tax. This value helps users identify which type of tax is applied.	Big, 62.0	62.0
taxableAmount	Double	Total amount of the line item that's eligible for taxation.	Big, 62.0	62.0

Tax Engine Log

Output representation of the logs that the tax engine generates.

JSON example

```
{
  "taxEngineLogs": [
    {
      "createdDate": "2022-03-09T10:55:38.000Z",
      "id": "3l1xx00000000PpAAI",
      "resultCode": "Success"
    }
  ]
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`createdDate`	String	Date when the tax engine creates the log.	Big, 62.0	62.0
`id`	String	ID of the tax engine log record.	Big, 62.0	62.0
`resultCode`	String	Result code associated with the created log.	Big, 62.0	62.0

Tax Imposition

Output representation of the details of the imposed tax.

JSON example

```
{
  "type": "Parish",
  "name": "Burbank"
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
`name`	String	Name of the tax imposition.	Big, 62.0	62.0
`type`	String	Type of the tax imposition.	Big, 62.0	62.0

Tax Jurisdiction

Output representation of the details of the tax jurisdiction for the tax line.

JSON example

```
{
    "country": "US",
    "id": "63000",
    "level": "CIT",
    "name": "SEATTLE",
    "region": "WA",
    "stateAssignedNo": "1726"
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
country	String	Country of the tax jurisdiction.	Big, 62.0	62.0
id	String	ID of the tax jurisdiction.	Big, 62.0	62.0
level	String	Level of the tax jurisdiction, for example, State and Federal.	Big, 62.0	62.0
name	String	Name of the tax jurisdiction authority.	Big, 62.0	62.0
region	String	Parent region of the tax jurisdiction.	Big, 62.0	62.0
stateAssignedNo	String	Number of the assigned state.	Big, 62.0	62.0

Billing Apex Reference

Billing provides the ConnectApi namespace (also called Connect in Apex) that contains classes for accessing the same capabilities that are available in the Billing Business APIs.

This table lists the available Billing Apex methods with the associated Connect REST API.

Apex Method	Connect REST API
convertNegativeInvoiceLines(ConvertNegativeInvoiceLinesInput, invoiceId)	Negative Invoice Lines to Credit Conversion (POST)
generateInvoices(inputRequest)	Invoices By Using Billing Schedules (POST)
recoverBillingSchedules(inputRequest)	Billing Schedule Recovery List (POST)
creditInvoice(CreditInvoiceInput, invoiceId)	Credit Memo Create and Apply (POST)
applyCreditMemos(CreditMemoApplyInput, creditMemoId)	Credit Memo Apply List(POST)
unapplyCreditMemos(CreditMemoUnapplyInput, creditMemoInvApplicationId)	Credit Memo Unapply (POST)
applyCreditMemoLines(CreditMemoLineApplyInput, creditMemoLineId)	Credit Memo Line Apply (POST)
unapplyCreditMemoLines(CreditMemoLineUnapplyInput, creditMemoLineInvoiceLineId)	Credit Memo Line Unapply (POST)
triggerInvoiceBatchDraftToPosted(invoiceBatchRunId)	Invoices Batch Draft to Posted Status (POST)
postDraftInvoices(inputRequest)	Invoice Draft to Posted Status (POST)

Apex Method	Connect REST API
`createCreditMemos(CreditMemoInputRequest)`	`Credit Memo Create and Apply (POST)`
`voidPostedInvoice(invoiceId)`	`Posted Invoice Voidance (POST)`
`calculateTax(calculateTax)`	`Tax Calculation (POST)`

ConnectApi Namespace
The `ConnectApi` namespace (also called Connect in Apex) provides classes to manage credit applications and billing scenarios.

TaxEngineAdapter Interface
Retrieves and evaluates the details from a tax engine to define tax details.

`ConnectApi` Namespace

The `ConnectApi` namespace (also called Connect in Apex) provides classes to manage credit applications and billing scenarios.

These classes are available in the `ConnectApi` namespace.

BatchInvoiceApplication Class
Update a batch of invoices from Draft to Posted status for a credit memo application. by using the BatchInvoiceApplication class.

Billing Class
Manage billing scenarios by using the Billing class. You can convert negative invoice lines, create and apply a credit memo to an invoice, generate invoices, and recover billing schedules.

CreditMemoApply Class
Manage credit memo applications by using the CreditMemoApply class.

CreditMemoLineApply Class
Manage credit memo line applications by using the CreditMemoLineApply class.

HarmonizeBilling Class
Update the status of the invoice from Draft to Posted by using the HarmonizeBilling class.

ConnectApi Input Classes
Billing includes these Apex input classes.

ConnectApi Output Classes
Billing includes these Apex output classes.

ConnectApi Enums
Enums specific to the `ConnectApi` namespace.

BatchInvoiceApplication Class

Update a batch of invoices from Draft to Posted status for a credit memo application. by using the BatchInvoiceApplication class.

Namespace

ConnectApi

BatchInvoiceApplication Methods

These methods are for `BatchInvoiceApplication`. All methods are static.

triggerInvoiceBatchDraftToPosted(invoiceBatchRunId)
Update a batch of invoices from Draft to Posted status for a credit memo application.

`triggerInvoiceBatchDraftToPosted(invoiceBatchRunId)`

Update a batch of invoices from Draft to Posted status for a credit memo application.

API Version

62.0

Requires Chatter

No

Signature

```
public static ConnectApi.InvoiceBatchDraftToPostedResult
triggerInvoiceBatchDraftToPosted(String invoiceBatchRunId)
```

Parameters

`invoiceBatchRunId`
Type: String

ID of the invoice batch run record that creates the draft invoices.

Return Value

Type: `ConnectApi.InvoiceBatchDraftToPostedResult` on page 1656

Usage

You need the Billing Operations User permission set to use this method.

This method posts the draft invoices and changes the status of the invoices from `Draft` to `Posted`.

Billing Class

Manage billing scenarios by using the Billing class. You can convert negative invoice lines, create and apply a credit memo to an invoice, generate invoices, and recover billing schedules.

Namespace

ConnectApi

Billing Methods

These methods are for `Billing`. All methods are static.

convertNegativeInvoiceLines(ConvertNegativeInvoiceLinesInput, invoiceId)
Convert a list of invoice lines with a negative amount into a posted credit memo. This conversion is applicable for a single invoice at a time.

createCreditMemos(CreditMemoInputRequest)
Create a credit memo without applying it to an invoice. You can credit the invoice at a later date.

creditInvoice(CreditInvoiceInput, invoiceId)
Create a credit memo and apply it to an invoice. The credit memo can fully or partially credit the invoice.

generateInvoices(inputRequest)
Create an invoice from a billing schedule.

recoverBillingSchedules(inputRequest)
Recover the latest generated invoice associated with the billing schedules in the Error or Processing status.

voidPostedInvoice(invoiceId)
Void a posted invoice to rebill the customer, if necessary.

`convertNegativeInvoiceLines(ConvertNegativeInvoiceLinesInput, invoiceId)`

Convert a list of invoice lines with a negative amount into a posted credit memo. This conversion is applicable for a single invoice at a time.

API Version

62.0

Requires Chatter

No

Signature

```
public static ConnectApi.ConvertNegativeInvoiceLinesResult
convertNegativeInvoiceLines(ConnectApi.ConvertNegativeInvoiceLinesInputRequest
ConvertNegativeInvoiceLinesInput, String invoiceId)
```

```
ConnectApi.Billing, convertNegativeInvoiceLines,
[ConnectApi.ConvertNegativeInvoiceLinesInputRequest, String],
ConnectApi.ConvertNegativeInvoiceLinesResult
```

Parameters

`ConvertNegativeInvoiceLinesInput`

Type: ConnectApi.ConvertNegativeInvoiceLinesInputRequest on page 1646

Input parameters to convert a negative invoice line to a credit.

`invoiceId`

Type: String

ID of the invoice whose negative invoice lines must be converted.

Return Value

Type: ConnectApi.ConvertNegativeInvoiceLinesResult on page 1656

Usage

You need the Credit Memo Operations User permission set to use this method.

Keep these considerations in mind when you use this method.

- All invoice lines must be related to the same invoice.
- The invoice line must have a negative amount.
- The invoice line must not be a previously converted credit memo.
- The invoice must have the `Posted` status.
- The invoice must not have any active settlements such as credit applications.

`createCreditMemos(CreditMemoInputRequest)`

Create a credit memo without applying it to an invoice. You can credit the invoice at a later date.

API Version

62.0

Requires Chatter

No

Signature

```
public static ConnectApi.RevenueAsyncRepresentation
createCreditMemos(ConnectApi.StandaloneCreditMemoInputRequest CreditMemoInputRequest)
```

Parameters

`CreditMemoInputRequest`

Type: `ConnectApi.StandaloneCreditMemoInputRequest` on page 1653

Input representation of the details required to create a standalone credit memo.

Return Value

Type: `ConnectApi.RevenueAsyncRepresentation` on page 1658

Usage

You need the Credit Memo Operations User permission set to use this method.

Specify the credit memo header information, charge parameters, adjustment parameters, and tax parameters. A credit memo requires at least one credit memo line. The credit memo line can be a charge or an adjustment.

Specify the credit memo lines that you want as lists of charges and adjustments. Each credit memo line must be related to a product.

`creditInvoice(CreditInvoiceInput, invoiceId)`

Create a credit memo and apply it to an invoice. The credit memo can fully or partially credit the invoice.

API Version

62.0

Requires Chatter

No

Signature

```
public static ConnectApi.RevenueAsyncLineLevelOutputResponse
creditInvoice(ConnectApi.CreditInvoiceInputRequest CreditInvoiceInput, String invoiceId)
```

```
ConnectApi.Billing, creditInvoice, [ConnectApi.CreditInvoiceInputRequest, String],
ConnectApi.RevenueAsyncLineLevelOutputResponse
```

Parameters

`CreditInvoiceInput`

Type: `ConnectApi.CreditInvoiceInputRequest` on page 1647

Input representation of the details of the request to create a credit memo.

`invoiceId`

Type: String

ID of the invoice to be credited partially or fully. The status of the invoice must be `Posted`.

Return Value

Type: `ConnectApi.RevenueAsyncLineLevelOutputResponse` on page 1657

Usage

Use this method to adjust an outstanding invoice balance or rectify errors in an invoice. Pass a list of invoice lines to credit. Keep these considerations in mind when you use this API.

- The request must contain at least one invoice line. Each invoice line must have the invoice line's ID, the amount to credit, and any optional tax details. The invoice lines must be a part of the invoice passed in the resource.
- The amount to credit must not exceed the charge or adjustment amount of an individual invoice line.
- The request body's credit amount inclusive of taxes must not exceed the target invoice line's amount inclusive of taxes, except for taxes calculated through an external tax service.
- The request body's total credit amount inclusive of taxes calculated through an external tax service must not exceed the outstanding invoice balance, which is also inclusive of taxes.

This method creates and posts a credit memo. The credit memo has one credit memo line for each invoice line passed in the API request. The invoice's balance is then reduced by a value equal to the credit memo's balance. This API modifies the balance of a posted invoice or invoice line based on the specified credit application level for your org. See Apply Credits to Posted Invoices or Invoice Lines.

`generateInvoices(inputRequest)`

Create an invoice from a billing schedule.

API Version

62.0

Requires Chatter

No

Signature

```
public static ConnectApi.RevenueAsyncRepresentation
generateInvoices(ConnectApi.InvoiceInputRepresentation inputRequest)
```

```
ConnectApi.Billing, generateInvoices, [ConnectApi.InvoiceInputRepresentation],
ConnectApi.RevenueAsyncRepresentation
```

Parameters

`inputRequest`

Type: ConnectApi.InvoiceInputRepresentation on page 1651

Input representation of the details of the billing schedule.

Return Value

Type: ConnectApi.RevenueAsyncRepresentation on page 1658

Usage

You need the Generate Invoices From Billing Schedule API permission set to use this method.

This method creates one billing period item for each unbilled period between the billing schedule's next billing date and the invoice's target date. Additionally, one invoice line is created for each billing period item. This method creates up to six billing periods per request.

`recoverBillingSchedules(inputRequest)`

Recover the latest generated invoice associated with the billing schedules in the Error or Processing status.

API Version

62.0

Requires Chatter

No

Signature

```
public static ConnectApi.BillingScheduleRecoveryResults
recoverBillingSchedules(ConnectApi.BillingScheduleRecoveryInputRequest inputRequest)
```

```
ConnectApi.Billing, recoverBillingSchedules,
[ConnectApi.BillingScheduleRecoveryInputRequest], ConnectApi.BillingScheduleRecoveryResults
```

Parameters

`inputRequest`

Type: ConnectApi.BillingScheduleRecoveryInputRequest on page 1646

Input representation of the details of the billing schedules to recover the associated invoice.

Return Value

Type: ConnectApi.BillingScheduleRecoveryResults on page 1656

Usage

You need the Manage Errors Using Invoice Error Recovery API permission set to use this method.

Billing schedules include critical details such as the amount to be billed, next billing date, and status. An invoice can be associated with one or more billing schedules. When an invoice is generated or posted, the billing schedules are updated to reflect the accurate state of the invoice. The billing schedules associated with an invoice are marked in the `Error` status if any of the invoicing processes have errors. Use this method to recover the invoice associated with the billing schedules in the `Error` or `Processing` status.

`voidPostedInvoice(invoiceId)`

Void a posted invoice to rebill the customer, if necessary.

API Version

62.0

Requires Chatter

No

Signature

```
public static ConnectApi.RevenueAsyncRepresentation voidPostedInvoice(String invoiceId)
```

Parameters

invoiceId
Type: String

ID of the posted invoice to be voided.

Return Value

Type: `ConnectApi.RevenueAsyncRepresentation` on page 1658

Usage

You need the Void a Posted Invoice API permission set to use this method.

This method changes the invoice status from `Posted` to `Void In Progress`. The invoice remains in the `Void In Progress` status until the credit is applied and financial fields are recalculated on the invoice's related billing period items and billing schedule. The invoice status changes to `Voided` after all recalculations are completed.

Keep these considerations in mind when you use this method.

- The balance and total amount on the invoice must be equal. If these amounts aren't equal due to payments or credits, the API request fails.
- You can't call other APIs on an invoice with the `Void In Progress` status. You also can't update the invoice fields.
- You can void only the most recently posted invoice on a billing schedule.
- To void an invoice that has payments or credits, use the Credit Memo Unapply (POST) API.

Credit Memos
The void process creates a credit memo, which contains one credit memo line for each invoice line, including tax lines. For example, if the invoice line has a balance of US$20, the related credit memo line has a balance of $20. The credit memo's balance is then allocated to the invoice header's balance, reducing it to zero. A credit memo invoice application is created to record the details of the void process.

Negative Invoice Lines
If an invoice has negative invoice lines that aren't converted to a credit memo, you can use this endpoint to void the posted invoice.

CreditMemoApply Class

Manage credit memo applications by using the CreditMemoApply class.

Namespace

ConnectApi

CreditMemoApply Methods

These methods are for `CreditMemoApply`. All methods are static.

applyCreditMemos(CreditMemoApplyInput, creditMemoId)

Adjust or correct already issued invoices by applying an existing credit memo to an invoice.

unapplyCreditMemos(CreditMemoUnapplyInput, creditMemoInvApplicationId)

Unapply a credit memo from an invoice and return the invoice and the credit memo to their pre-application states.

applyCreditMemos(CreditMemoApplyInput, creditMemoId)

Adjust or correct already issued invoices by applying an existing credit memo to an invoice.

API Version

62.0

Requires Chatter

No

Signature

```
public static ConnectApi.ApplyCreditResults
applyCreditMemos(ConnectApi.CreditMemoApplyInputRequest CreditMemoApplyInput, String
creditMemoId)
```

Parameters

CreditMemoApplyInput

Type: ConnectApi.CreditMemoApplyInputRequest on page 1650

Input representation of the request to apply a credit memo to an invoice.

creditMemoId

Type: String

ID of the credit memo record.

Return Value

Type: ConnectApi.ApplyCreditResults on page 1655

Usage

Specify the credit memo ID and the amounts to be applied, with the total of all applied amounts not exceeding the credit memo's balance.

The credit amount for each invoice can't surpass the original charge or adjustment amount, and the overall credit amount must not exceed the invoice's outstanding balance. The exceptions include any taxes calculated by an external service.

For example, your organization sold 10 tablets at US$500 each, totaling $5000, to a vendor who later reported that 6 tablets were defective. Using this method, your accounts receivable team creates a $3000 credit memo and applies this credit to the original invoice.

unapplyCreditMemos(CreditMemoUnapplyInput, creditMemoInvApplicationId)

Unapply a credit memo from an invoice and return the invoice and the credit memo to their pre-application states.

API Version

62.0

Requires Chatter

No

Signature

```
public static ConnectApi.UnapplyCreditResult
unapplyCreditMemos(ConnectApi.CreditMemoUnapplyInputRequest CreditMemoUnapplyInput,
String creditMemoInvApplicationId)
```

Parameters

CreditMemoUnapplyInput
Type: ConnectApi.CreditMemoUnapplyInputRequest on page 1650

Input representation of the request to unapply a credit memo from an invoice.

creditMemoInvApplicationId
Type: String

ID of the credit memo invoice application.

Return Value

Type: ConnectApi.UnapplyCreditResult on page 1658

Usage

Use this method if an error occurred when a credit is issued. For example, if an incorrect credit memo is applied to an invoice, or if a credit memo is created for an incorrect amount, use this method to unapply the credit memo.

CreditMemoLineApply Class

Manage credit memo line applications by using the CreditMemoLineApply class.

Namespace

ConnectApi

CreditMemoLineApply Methods

These methods are for `CreditMemoLineApply`. All methods are static.

applyCreditMemoLines(CreditMemoLineApplyInput, creditMemoLineId)
Adjust or correct already issued invoices by applying an existing credit memo line to an invoice line.

unapplyCreditMemoLines(CreditMemoLineUnapplyInput, creditMemoLineInvoiceLineId)
Unapply a credit memo line from an invoice line and return the invoice line and the credit memo line to their pre-application states.

applyCreditMemoLines(CreditMemoLineApplyInput, creditMemoLineId)

Adjust or correct already issued invoices by applying an existing credit memo line to an invoice line.

API Version

62.0

Requires Chatter

No

Signature

```
public static ConnectApi.CreditMemoLineAppliedResponse
applyCreditMemoLines(ConnectApi.CreditMemoLineApplyInput CreditMemoLineApplyInput,
String creditMemoLineId)
```

Parameters

CreditMemoLineApplyInput
Type: ConnectApi.CreditMemoLineApplyInput on page 1650

Input representation of the details of the request to apply a credit memo line to an invoice line.

creditMemoLineId
Type: String

ID of the credit memo line record.

Return Value

Type: ConnectApi.CreditMemoLineAppliedResponse on page 1656

Usage

You need the Credit Memo Operations User permission set to use this method.

unapplyCreditMemoLines(CreditMemoLineUnapplyInput, creditMemoLineInvoiceLineId)

Unapply a credit memo line from an invoice line and return the invoice line and the credit memo line to their pre-application states.

API Version

62.0

Requires Chatter

No

Signature

```
public static ConnectApi.CreditMemoLineUnappliedResponse
unapplyCreditMemoLines(ConnectApi.CreditMemoLineUnapplyInput CreditMemoLineUnapplyInput,
String creditMemoLineInvoiceLineId)
```

Parameters

CreditMemoLineUnapplyInput
Type: ConnectApi.CreditMemoLineUnapplyInput on page 1650

Input representation of the details of the request to unapply a credit memo line from an invoice line.

creditMemoLineInvoiceLineId
Type: String

ID of the credit memo line invoice line record.

Return Value

Type: ConnectApi.CreditMemoLineUnappliedResponse on page 1656

Usage

You need the Credit Memo Operations User permission set to use this method.

HarmonizeBilling Class

Update the status of the invoice from Draft to Posted by using the HarmonizeBilling class.

Namespace

ConnectApi

HarmonizeBilling Methods

These methods are for `HarmonizeBilling`. All methods are static.

postDraftInvoices(inputRequest)
Update the status of the invoice from Draft to Posted.

`postDraftInvoices(inputRequest)`

Update the status of the invoice from Draft to Posted.

API Version

62.0

Requires Chatter

No

Signature

```
public static ConnectApi.RevenueAsyncRepresentation
postDraftInvoices(ConnectApi.InvoiceDraftToPostedInputRequest inputRequest)
```

Parameters

inputRequest

Type: `ConnectApi.InvoiceDraftToPostedInputRequest` on page 1651

Input representation of the details of the draft invoice that's posted.

Return Value

Type: `ConnectApi.RevenueAsyncRepresentation` on page 1658

Usage

You need the Billing Operations User permission set to use this method.

This method calls an external tax engine to calculate taxes for the draft invoice, posts the invoice, and updates the related billing schedules and billing periods.

ConnectApi Input Classes

Billing includes these Apex input classes.

ConnectApi.ApplicationsRequest
Connect API representation of an application item input request for credit memo apply api

ConnectApi.BillingAddressRequest
Input representation of the details of an address.

ConnectApi.BillingScheduleRecoveryInputRepresentation
Input representation of the details of the billing schedules to recover the associated invoice.

ConnectApi.ConvertNegativeInvoiceLinesInputRequest
Input representation of the request details to convert a negative invoice line into a credit.

ConnectApi.CreditDetailsApplyInput
Input representation of the request to specify one or more applications to apply a credit memo line for, with each application representing an invoice line.

ConnectApi.CreditInvoiceInputRequest
Input representation of the details of the request to create a credit memo.

ConnectApi.CreditInvoiceInvoiceLine
Input representation of the details of the invoice lines to be credited.

ConnectApi.CreditInvoiceInvoiceLineTax
Input representation of the details of the tax lines to be created manually for the invoice line.

ConnectApi.CreditMemoAddressesInputRequest
Input representation of the details of the billing and shipping addresses.

ConnectApi.CreditMemoApplyInputRequest
Input representation of the request to apply a credit memo to an invoice.

ConnectApi.CreditMemoLineApplyInput
Input representation of the details of the request to apply a credit memo line to an invoice line.

ConnectApi.CreditMemoLineUnapplyInput
Input representation of the details of the request to unapply a credit memo line from an invoice line.

ConnectApi.CreditMemoUnapplyInputRequest
Input representation of the request to unapply a credit memo from an invoice.

ConnectApi.InvoiceDraftToPostedInputRequest
Input representation of the details of the draft invoice that's posted.

ConnectApi.InvoiceInputRepresentation
Input representation of the details of the billing schedule.

ConnectApi.StandaloneCreditMemoChargeInputRequest
Input representation of the details of the charge lines of a credit memo.

ConnectApi.StandaloneCreditMemoInputRequest
Input representation of the details required to create a standalone credit memo.

ConnectApi.StandaloneCreditMemoTaxInputRequest
Connect API representation of Tax input request

ConnectApi.ApplicationsRequest

Connect API representation of an application item input request for credit memo apply api

Property	Type	Description	Required or Optional	Available Version
`amount`	Double	Credit amount to be applied to the invoice.	Required	62.0
`appliedToId`	String	ID of the invoice record to apply the credit for.	Required	62.0
`description`	String	Explanation or reason for applying the credit memo.	Optional	62.0
`effectiveDate`	String	Effective date for the credit memo.	Optional	62.0

ConnectApi.BillingAddressRequest

Input representation of the details of an address.

Property	Type	Description	Required or Optional	Available Version
`city`	String	Address city.	Optional	62.0
`country`	String	Address country.	Optional	62.0

Property	Type	Description	Required or Optional	Available Version
latitude	Double	Latitude for the address.	Optional	62.0
locationCode	String	Location code for the address.	Optional	62.0
longitude	Double	Longitude for the address.	Optional	62.0
postalCode	String	Postal code for the address.	Optional	62.0
state	String	Address state.	Optional	62.0
street	String	Address street.	Optional	62.0

ConnectApi.BillingScheduleRecoveryInputRepresentation

Input representation of the details of the billing schedules to recover the associated invoice.

Property	Type	Description	Required or Optional	Available Version
billingScheduleIds	List<String>	IDs of the billing schedules to recover the invoice for. You can recover one billing schedule per API request.	Required	62.0

ConnectApi.ConvertNegativeInvoiceLinesInputRequest

Input representation of the request details to convert a negative invoice line into a credit.

Property	Type	Description	Required or Optional	Available Version
description	String	Description stamped on the credit memo that's created after the negative invoice line conversion.	Optional	62.0
effectiveDate	String	Date stamped on the credit memo that's created after the negative invoice line conversion.	Required	62.0
invoiceId	String	ID of the invoice whose negative invoice lines must be converted.	Required	62.0
invoiceLines	List<String>	List of the negative invoice lines along with the associated invoice line taxes.	Required	62.0

ConnectApi.CreditDetailsApplyInput

Input representation of the request to specify one or more applications to apply a credit memo line for, with each application representing an invoice line.

Property	Type	Description	Required or Optional	Available Version
`appliedAmount`	Double	Credit amount to be applied to the invoice line.	Required	62.0
`description`	String	Explanation or reason for applying the credit memo line.	Optional	62.0
`effectiveDate`	String	Effective date for the credit memo line.	Optional	62.0
`invoiceLineId`	String	ID of the invoice line record to apply the credit for.	Required	62.0

ConnectApi.CreditInvoiceInputRequest

Input representation of the details of the request to create a credit memo.

Property	Type	Description	Required or Optional	Available Version
`description`	String	Description for the credit memo to be created.	Optional	62.0
`effectiveDate`	String	Date when the credit memo takes effect.	Optional	62.0
`invoiceLines`	List<ConnectApi.CreditInvoiceLine>	List of the invoice lines to be credited. The invoice line IDs must be related to the invoice ID specified in the API request. If invoice lines aren't specified, the API request results in an error.	Required	62.0
`taxEffectiveDate`	String	Date when the tax takes effect to recalculate the taxes.	Optional	62.0
`taxStrategy`	ConnectApi.TaxStrategyEnum	Tax strategy to be applied across invoice lines. You can override the tax strategy at the individual invoice line level or at the tax line level. Valid values are: • `Ignore`—Specifies that the creation of tax lines must be ignored. • `CopyFromInvoiceLine`—Specifies that tax values must be copied from the invoice line.	Required	62.0

Property	Type	Description	Required or Optional	Available Version
		• `Calculate`—Specifies that tax must be calculated by using the API.		
`type`	ConnectApi.CreditMemoTypeEnum	Type of credit memo to be created. Valid value is `Posted`.	Optional	62.0

ConnectApi.CreditInvoiceInvoiceLine

Input representation of the details of the invoice lines to be credited.

Property	Type	Description	Required or Optional	Available Version
`addresses`	ConnectApi.CreditMemoAddressInput	Addresses to be created manually for this invoice line and the overridden tax lines. These addresses are only applicable if this invoice line is using the `ManualOverride` tax strategy.	Optional	62.0
`amountToCredit`	Double	Amount to be credited from this invoice line.	Required	62.0
`invoiceLineId`	String	ID of the invoice line record to be credited. The invoice line ID must be related to the invoice ID specified in the API request.	Required	62.0
`isTaxOnlyCredit`	Boolean	Indicates whether the applicable tax amount is credited for the charge or adjustment amount (`true`), or the applicable tax amount is credited along with the charge or adjustment amount (`false`). The default value is `false`.	Optional	62.0
`taxEffectiveDate`	String	Date when the tax takes effect and the invoice line is credited.	Optional	62.0
`taxStrategy`	ConnectApi.TaxStrategyEnum	Tax strategy for crediting the invoice line. This tax strategy takes precedence over the `taxStrategy` property value specified in the Credit Invoice Input. Valid values are: • `Ignore`—Specifies that the creation of tax lines must be ignored.	Optional	62.0

Property	Type	Description	Required or Optional	Available Version
		• `CopyFromInvoiceLine`—Specifies that tax values must be copied from the invoice line. • `Calculate`—Specifies that tax must be calculated by using the API.		
`taxes`	List<ConnectApi.CreditInvoiceInvoiceLineTax>	List of tax lines to be created manually for this invoice line.	Required if the `taxStrategy` property value is `ManualOverride`.	62.0

ConnectApi.CreditInvoiceInvoiceLineTax

Input representation of the details of the tax lines to be created manually for the invoice line.

Property	Type	Description	Required or Optional	Available Version
`taxAmount`	Double	Amount of tax to be applied related to this invoice line.	Required	62.0
`taxCode`	String	Tax code to be applied related to this invoice line to create the tax line.	Optional	62.0
`taxName`	String	Name of tax to be applied related to this invoice line.	Optional	62.0
`taxRate`	Double	Tax rate used to create the tax line.	Optional	62.0

ConnectApi.CreditMemoAddressesInputRequest

Input representation of the details of the billing and shipping addresses.

Property	Type	Description	Required or Optional	Available Version
`billingAddress`	ConnectApi.BillingAddressRequest	Billing address for charge or adjustment line.	Optional	62.0
`shippingAddress`	ConnectApi.BillingAddressRequest	Shipping address for charge or adjustment line.	Optional	62.0

ConnectApi.CreditMemoApplyInputRequest

Input representation of the request to apply a credit memo to an invoice.

Property	Type	Description	Required or Optional	Available Version
applications	List<ConnectApi.ApplicationRequest>	List of one or more applications to apply the credit memo for. Each application represents an invoice that's credited by using the balance of the specified credit memo.	Required	62.0
creditMemoId	String	ID of the credit memo record.	Required	62.0

ConnectApi.CreditMemoLineApplyInput

Input representation of the details of the request to apply a credit memo line to an invoice line.

Property	Type	Description	Required or Optional	Available Version
applyCreditDetails	List<ConnectApi.CreditDetailApplyInput>	List of one or more applications to apply the credit memo line for. Each application represents an invoice line that's credited by using the balance of the specified credit memo line.	Required	62.0
creditMemoLineId	String	ID of the credit memo line record.	Required	62.0

ConnectApi.CreditMemoLineUnapplyInput

Input representation of the details of the request to unapply a credit memo line from an invoice line.

Property	Type	Description	Required or Optional	Available Version
creditMemoLineInvoiceLineId	String	ID of the credit memo line invoice line record.	Required	62.0
description	String	Explanation or reason for unapplying the credit memo line.	Optional	62.0
effectiveDate	String	Effective date for the credit memo line.	Optional	62.0

ConnectApi.CreditMemoUnapplyInputRequest

Input representation of the request to unapply a credit memo from an invoice.

Property	Type	Description	Required or Optional	Available Version
creditMemoInvApplicationId	String	ID of the credit memo invoice application.	Required	62.0
description	String	Explanation or reason for unapplying the credit memo.	Optional	62.0
effectiveDate	String	Effective date for the credit memo.	Optional	62.0

ConnectApi.InvoiceDraftToPostedInputRequest

Input representation of the details of the draft invoice that's posted.

Property	Type	Description	Required or Optional	Available Version
correlationId	String	Splunk correlation ID to track the messages that are related to the request and are logged in Splunk by the different services involved in the request. If the ID isn't specified, the service creates a random Universally Unique Identifier (UUID).	Optional	62.0
invoiceIds	List<String>	IDs of the invoice records in Draft status to be posted. You can post one draft invoice per API request.	Required	62.0

ConnectApi.InvoiceInputRepresentation

Input representation of the details of the billing schedule.

Property	Type	Description	Required or Optional	Available Version
action	ConnectApi.InvoiceAction	Type of invoice to be created. Valid values are: • Draft • Posted	Required	62.0
billingScheduleIds	List<String>	List of billing schedule IDs that's used to create the invoices. You can specify a maximum of 200 billing schedules.	Required	62.0
correlationId	String	Property that's tagged against the published InvoiceProcessedEvent event, if specified.	Optional	62.0
invoiceDate	String	Stamping date of the invoice in ISO 8601 format.	Required	62.0

Property	Type	Description	Required or Optional	Available Version
`targetDate`	String	Date in ISO 8601 format used to decide the billing periods that are included to create invoices.	Required	62.0

ConnectApi.StandaloneCreditMemoChargeInputRequest

Input representation of the details of the charge lines of a credit memo.

Property	Type	Description	Required or Optional	Available Version
`addresses`	Credit Memo Addresses Input	Details of the billing and shipping addresses.	Optional	62.0
`chargeAmount`	Double	Charge amount for the credit memo.	Required	62.0
`description`	String	Description of the created credit memo charge line.	Optional	62.0
`endDate`	String	End date of the credit memo charge line.	Optional	62.0
`isTaxOnlyCredit`	Boolean	Indicates whether the credit is for tax only (`true`) or not (`false`).	Optional	62.0
`productId`	String	ID of the product record that the credit memo is issued on.	Optional	62.0
`productName`	String	Name of the product that the credit memo is issued on.	Optional	62.0
`startDate`	String	Start date of the credit memo charge line.	Optional	62.0
`taxEffectiveDate`	String	Date from when the tax is applicable.	Optional	62.0
`taxes`	[illegible]	List of credit memo tax lines.	Optional	62.0
`taxStrategy`	[illegible]	Tax strategy to be applied to this credit memo charge line, child treatment lines, and tax lines. You can override the tax strategy at the individual credit memo lines or tax lines level. Valid values are: • `Ignore`—Specifies that the creation of tax lines must be ignored. • `Calculate`—Specifies that tax must be calculated by using the API.	Optional	62.0
`treatmentId`	String	ID of the tax treatment record that's used to calculate tax.	Optional	62.0

ConnectApi.StandaloneCreditMemoInputRequest

Input representation of the details required to create a standalone credit memo.

Property	Type	Description	Required or Optional	Available Version
`billToContactId`	String	Contact related to the credit memo.	Optional	62.0
`billingAccountId`	String	ID of the account that the credit is issued to.	Required	62.0
`charges`	[illegible]	Charge lines of the credit memo. Requires at least one charge line.	Required	62.0
`currencyIsoCode`	String	ISO code currency of the new credit that's issued.	Optional	62.0
`description`	String	Description for the new credit that's issued.	Optional	62.0
`effectiveDate`	String	Effective date of the credit memo. If the value isn't specified, then it's null.	Optional	62.0
`externalReference`	String	ID of the external reference for the credit memo.	Optional	62.0
`externalReferenceDataSource`	String	Source of the external reference for the credit memo.	Optional	62.0
`taxEffectiveDate`	String	Effective date of the credit memo tax. If the value isn't specified, then it's null.	Optional	62.0
`taxStrategy`	[illegible]	Specifies how tax lines must be created for the standalone credit memos. Valid values are: • `Ignore`—Specifies that the creation of tax lines must be ignored. • `Calculate`—Specifies that tax must be calculated by using the API.	Required	62.0
`type`	[illegible]	Type of credit memo to be created. Valid value is `Posted`.	Optional	62.0

ConnectApi.StandaloneCreditMemoTaxInputRequest

Connect API representation of Tax input request

Property	Type	Description	Required or Optional	Available Version
`taxAmount`	Double	Amount of tax to be applied.	Required	62.0
`taxCode`	String	Tax code to be used to create the tax line.	Optional	62.0

Property	Type	Description	Required or Optional	Available Version
`taxName`	String	Name of tax to be applied.	Optional	62.0
`taxRate`	Double	Tax rate to be used to create the tax line.	Optional	62.0

ConnectApi Output Classes

Billing includes these Apex output classes.

ConnectApi.AppliedCreditResponse
Output representation of the list of applied credit memo line results.

ConnectApi.ApplyCreditResults
Output representation of the list of applied credit memo results.

ConnectApi.ApplyCreditResult
Connect API representation of credit memo apply output result

ConnectApi.BillingScheduleRecoveryResult
Output representation of the details of the recovered billing schedules.

ConnectApi.BillingScheduleRecoveryResults
Output representation of the recovered details of the billing schedules and associated invoice.

ConnectApi.CreditMemoLineAppliedResponse
Output representation of the list of applied credit memo line results.

ConnectApi.CreditMemoLineUnappliedResponse
Output representation of the details of the credit memo line invoice line record with the status of the request.

ConnectApi.ConvertNegativeInvoiceLinesResult
Output representation of the details of the credit memo along with the status of the request.

ConnectApi.InvoiceBatchDraftToPostedResult
Output representation of the batch update details of the invoices from `Draft` to `Posted` status.

ConnectApi.InvoiceRecoveryResult
Output representation of the details of the recovered invoice and billing schedules.

ConnectApi.ReferenceLineError
Output representation of the details of the line level errors.

ConnectApi.RevenueAsyncLineLevelOutputResponse
Output representation of the result of the API request for the async line level operations.

ConnectApi.RevenueAsyncRepresentation
Output representation of the result of the API request with the request identifier.

ConnectApi.UnapplyCreditResult
Output representation of the details of the credit memo invoice application record with the status of the request.

ConnectApi.AppliedCreditResponse

Output representation of the list of applied credit memo line results.

Property Name	Type	Description	Available Version
creditMemoLineInvoiceLineId	String	ID of the credit memo line invoice line ID.	62.0
errors	List<ConnectApi.ErrorResponse>	List of errors encountered during the processing of the API request.	62.0
invoiceLineId	String	ID of the invoice line record that the credit is applied to.	62.0
success	Boolean	Indicates whether the credit memo line is successfully applied (true) or not (false).	62.0

ConnectApi.ApplyCreditResults

Output representation of the list of applied credit memo results.

Property Name	Type	Description	Available Version
applyCreditResults	List<ConnectApiApplyCreditResult>	Output list of the applied credit memo results.	62.0

ConnectApi.ApplyCreditResult

Connect API representation of credit memo apply output result

Property Name	Type	Description	Available Version
appliedToId	String	ID of the invoice record that the credit is applied to.	62.0
errors	List<ConnectApiErrorResponse>	List of errors encountered during the processing of the API request.	62.0
id	String	ID of the credit memo invoice application.	62.0
success	Boolean	Indicates whether the credit memo is successfully applied (true) or not (false).	62.0

ConnectApi.BillingScheduleRecoveryResult

Output representation of the details of the recovered billing schedules.

Property Name	Type	Description	Available Version
billingScheduleId	String	ID of the billing schedule.	62.0
billingScheduleStatus	String	Flag that indicates the billing schedule status.	62.0

ConnectApi.BillingScheduleRecoveryResults

Output representation of the recovered details of the billing schedules and associated invoice.

Property Name	Type	Description	Available Version
recoveryResults	List<ConnectApiInvoiceRecoveryResult>	Details of the recovered invoice and billing schedules.	62.0

ConnectApi.CreditMemoLineAppliedResponse

Output representation of the list of applied credit memo line results.

Property Name	Type	Description	Available Version
appliedCreditResponses	List<ConnectApi.AppliedCreditResponse>	Output list for the applied credit memo line results.	62.0

ConnectApi.CreditMemoLineUnappliedResponse

Output representation of the details of the credit memo line invoice line record with the status of the request.

Property Name	Type	Description	Available Version
creditMemoLineInvoiceLineId	String	ID of the credit memo line invoice line record.	62.0
errors	List<ConnectApi.ErrorResponse>	List of errors encountered during the processing of the API request.	62.0
success	Boolean	Indicates whether the credit memo line is successfully unapplied (`true`) or not (`false`).	62.0

ConnectApi.ConvertNegativeInvoiceLinesResult

Output representation of the details of the credit memo along with the status of the request.

Property Name	Type	Description	Available Version
errors	List<ConnectApiErrorResponse>	List of errors encountered during the processing of the API request.	62.0
id	String	ID of the credit memo that's created after the conversion of the negative invoice lines.	62.0
success	Boolean	Indicates whether the API request is successful (`true`) or not (`false`).	62.0

ConnectApi.InvoiceBatchDraftToPostedResult

Output representation of the batch update details of the invoices from `Draft` to `Posted` status.

Property Name	Type	Description	Available Version
invoiceBatchDraftToPostedId	String	ID of the invoice batch draft to posted run record that's created to track the batch process of posting the draft invoices that are associated with the parent invoice batch run record.	62.0
success	Boolean	Indicates whether the API request is successful (true) or not (false).	62.0

ConnectApi.InvoiceRecoveryResult

Output representation of the details of the recovered invoice and billing schedules.

Property Name	Type	Description	Available Version
billingSchedules	List<ConnectApi.BillingScheduleRecoveryResult>	Billing schedules associated with this invoice.	62.0
invoiceErrors	List<ConnectApi.ErrorResponse>	List of errors encountered during the invoice recovery.	62.0
invoiceId	String	ID of the recovered invoice.	62.0
invoiceStatus	String	Flag that indicates the invoice status.	62.0
success	Boolean	Indicates whether the overall transaction was successful or not (true) or not (false).	62.0

ConnectApi.ReferenceLineError

Output representation of the details of the line level errors.

Property Name	Type	Description	Available Version
errors	List<ConnectApi.ErrorResponse>	List of errors with error code and error message for the specified invoice line ID.	62.0
referenceLineId	String	ID of the invoice line specified in the API request that has an issue, causing the API request to fail.	62.0

ConnectApi.RevenueAsyncLineLevelOutputResponse

Output representation of the result of the API request for the async line level operations.

Property Name	Type	Description	Available Version
errors	List<ConnectApi.ErrorResponse>	Details of errors, if any.	62.0

Property Name	Type	Description	Available Version
referenceLineErrorResults	List<ConnectApiReferenceLineError>	List of errors grouped by the invoice line IDs if the API request fails.	62.0
referenceLineType	String	Reference type for the reference line entity in the `referenceLineErrorResults` property.	62.0
requestIdentifier	String	Unique identifier of the request.	62.0
statusURL	String	URL to track the status of the operation.	62.0
success	Boolean	Indicates whether the API request is successful (`true`) or not (`false`).	62.0

ConnectApi.RevenueAsyncRepresentation

Output representation of the result of the API request with the request identifier.

Property Name	Type	Description	Available Version
errors	List<ConnectApiErrorResponse>	Details of errors, if any.	62.0
request Identifier	String	Unique identifier of the request.	62.0
success	Boolean	Indicates whether the API request is successful (`true`) or not (`false`).	62.0

ConnectApi.UnapplyCreditResult

Output representation of the details of the credit memo invoice application record with the status of the request.

Property Name	Type	Description	Available Version
errors	List<ConnectApiErrorResponse>	List of errors encountered during the processing of the API request.	62.0
id	String	ID of the credit memo invoice application record.	62.0
success	Boolean	Indicates whether the credit memo is successfully unapplied (`true`) or not (`false`).	62.0

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.

Enum	Description
`ConnectApi.InvoiceAction`	Type of invoice to be created. Valid values are: • `Draft` • `Posted`
`ConnectApi.TaxStrategyEnum`	Tax strategy to be applied across invoice lines. You can override the tax strategy at the individual invoice line level or at the tax line level. Valid values are: • `Ignore`—Specifies that the creation of tax lines must be ignored. • `CopyFromInvoiceLine`—Specifies that tax values must be copied from the invoice line. • `Calculate`—Specifies that tax must be calculated by using the API.
`ConnectApi.CreditMemoTypeEnum`	Type of credit memo to be created. Valid value is `Posted`.
`ConnectApi.StandaloneTaxStrategyEnum`	Specifies how tax lines must be created for the standalone credit memos. Valid values are: • `Ignore`—Specifies that the creation of tax lines must be ignored. • `Calculate`—Specifies that tax must be calculated by using the API.

TaxEngineAdapter Interface

Retrieves and evaluates the details from a tax engine to define tax details.

You can extend the TaxEngineAdapter interface to define a custom tax adapter based on your requirements. Use the custom tax adapter with Billing services to implement standard tax.

Create a custom object and associated fields to store tax details, such as tax rate for a country. For example, create a custom object named CountryTaxRate with Country_Code and Tax_Rate fields. Create records to define the details for these fields.

TaxEngineAdapter Methods

Learn more about the available methods with the `TaxEngineAdapter` class.

TaxEngineAdapter Example Implementation

Refer to the example implementation of the `TaxEngineAdapter` interface to accept information from a tax engine and evaluate the information to define tax details.

SEE ALSO:

Salesforce Help: Tax Calculation for Invoices

Billing Business APIs: Tax Calculation (POST)

Billing Standard Objects: TaxEngineProvider, TaxEngine, TaxPolicy, and TaxTreatment

TaxEngineAdapter Methods

Learn more about the available methods with the `TaxEngineAdapter` class.

The `TaxEngineAdapter` class includes these methods.

processRequest(requestType)
The `processRequest` method takes an instance of `TaxEngineContext` class and returns a response with the calculated tax details through the `TaxDetailsResponse` class or an error response through the `ErrorResponse` class.

`processRequest(requestType)`

The `processRequest` method takes an instance of `TaxEngineContext` class and returns a response with the calculated tax details through the `TaxDetailsResponse` class or an error response through the `ErrorResponse` class.

Signature

```
global commercetax.TaxEngineResponse processRequest(commercetax.TaxEngineContext var1)
```

Parameters

var1
Type: TaxEngineContext

Wrapper class that stores information about the type of a tax calculation request.

Return Value

Type: TaxEngineResponse

Generic interface representing a response from a tax engine.

TaxEngineAdapter Example Implementation

Refer to the example implementation of the `TaxEngineAdapter` interface to accept information from a tax engine and evaluate the information to define tax details.

Namespace

See commercetax namespace to view the list of available classes.

Usage

The `TaxEngineAdapter` interface accepts information from the tax engine through the `TaxEngineContext` class. The interface evaluates the information to define tax in the response with details, such as tax amount and addresses. The response is used to update and create entities in the Salesforce org.

Use these steps to build a sample tax adapter implementation. Each tax adapter implementation varies based on your implementation requirements. Customize this example to suit your business requirements.

Example:

- Get the tax rates from the created custom object. See TaxEngineAdapter Interface on page 1659.

- The custom adapter class implements the `TaxEngineAdapter` interface. The `processRequest` method takes an instance of `TaxEngineContext` class and returns a response with the calculated tax details through the `TaxDetailsResponse` class or an error response through the `ErrorResponse` class.

```
global virtual class StandardTaxAdapter implements commercetax.TaxEngineAdapter {

    global commercetax.TaxEngineResponse processRequest(commercetax.TaxEngineContext
 taxEngineContext) {
        commercetax.RequestType requestType = taxEngineContext.getRequestType();
        commercetax.CalculateTaxRequest request =
(commercetax.CalculateTaxRequest)taxEngineContext.getRequest();
        if(request.documentCode == null) {
            return new
commercetax.ErrorResponse(commercetax.resultcode.TaxEngineError, '404', 'documentCode
  is mandatory');
        }
        if(request.documentCode == 'TaxEngineError') {
            return new
commercetax.ErrorResponse(commercetax.resultcode.TaxEngineError, '504', 'documentCode
  - not supported');
        }
        if(request.documentCode == 'simulateValidationFailureInAdapter') {
            return new
commercetax.ErrorResponse(commercetax.resultcode.TaxEngineError, '400', 'validations
 for documentCode failed in adapter');
        }
        if(request.documentCode == 'simulateMalformedErrorInAdapter') {
                    return new
commercetax.ErrorResponse(commercetax.resultcode.TaxEngineError, null, 'malformed
adapter error response');
        }
        if(request.documentCode == 'simulateTaxEngineProcessFailure') {
            return new
commercetax.ErrorResponse(commercetax.resultcode.TaxEngineError, '500', 'Tax Engine
 couldnt process your request');
        }

        if(requestType == commercetax.RequestType.CalculateTax){
            commercetax.calculatetaxtype type = request.taxtype;
            String docCode;
            if (request.getDocumentCode() != null) {
                docCode = request.getDocumentCode();
            } else {
                docCode = String.valueOf(getRandomInteger(0, 2147483647));
            }
            commercetax.CalculateTaxResponse response = new
commercetax.CalculateTaxResponse();
            if(request.isCommit == true) {
                response.setStatus(commercetax.TaxTransactionStatus.Committed);
            } else {
                response.setStatus(commercetax.TaxTransactionStatus.Uncommitted);
            }
            response.setDocumentCode(docCode);
            response.setReferenceDocumentCode(request.referenceDocumentCode);
```

```
            response.setTaxType(type);
            response.setStatusDescription('statusDescription');
            if(request.sellerDetails.code == 'testSellerCode') {
               response.setDescription('SellerCode fetched from TaxEngine entity');

            }
            else {
                response.setDescription('description');
            }
           response.setEffectiveDate(system.now());
           if(request.transactionDate == null) {
              response.setTransactionDate(system.now());
           } else {
              response.setTransactionDate(request.transactionDate);
           }
           if(request.taxTransactionType == null) {
             response.setTaxTransactionType(commercetax.TaxTransactionType.Debit);

           } else {
               response.setTaxTransactionType(request.taxTransactionType);
           }

            if(request.currencyIsoCode == null || request.currencyIsoCode == '') {
               response.setCurrencyIsoCode('USD');
            } else {
               response.setCurrencyIsoCode(request.currencyIsoCode);
            }
            response.setReferenceEntityId(request.ReferenceEntityId);

            Map<String, Double> countryTaxRateMap = getTaxes(request.lineItems);

            Double totalTax = 0.0;
            Double totalAmount = 0.0;
            List<commercetax.LineItemResponse> lineItemResponses = new
List<commercetax.LineItemResponse>();
            for(Commercetax.TaxLineItemRequest lineItem : request.lineItems){
                String country = getCountryFromLineItem(lineItem);
                if (country == null) {
                    return new
commercetax.ErrorResponse(commercetax.resultcode.TaxEngineError, '400', 'Country is
 mandatory for each line item');
                }

                Double taxRate = countryTaxRateMap.get(country);
                if (taxRate == null) {
                    return new
commercetax.ErrorResponse(commercetax.resultcode.TaxEngineError, '404', 'No tax rate
 found for the specified country: ' + country);
                }
                commercetax.AddressesResponse addressesRes = new
commercetax.AddressesResponse();
                if(request.DocumentCode == 'SetsNullForResponseWithoutException'){
                    addressesRes.setShipFrom(null);
                    addressesRes.setShipTO(null);
```

```
                    addressesRes.setSoldTo(null);
                }else{
                    commercetax.AddressResponse addRes = new
commercetax.AddressResponse();
                    addRes.setLocationCode('locationCode');
                    addressesRes.setShipFrom(addRes);
                    addressesRes.setShipTO(addRes);
                    addressesRes.setSoldTo(addRes);
                }
                commercetax.LineItemResponse lineItemResponse = new
commercetax.LineItemResponse();
                Double totalLineTax = 0;
                List<commercetax.TaxDetailsResponse> taxDetailsResponses = new
List<commercetax.TaxDetailsResponse>();

                Double taxableAmount = lineItem.amount;
                commercetax.TaxDetailsResponse taxDetailsResponse = new
commercetax.TaxDetailsResponse();
                taxDetailsResponse.setRate(Double.valueOf(taxRate));
                taxDetailsResponse.setTaxableAmount(taxableAmount);
                Double tax = taxableAmount*taxRate;
                totalLineTax+=tax;
                taxDetailsResponse.setTax(taxableAmount*taxRate);
                taxDetailsResponse.setExemptAmount(0);
                taxDetailsResponse.setExemptReason('exemptReason');
                taxDetailsResponse.setTaxRegionId('taxRegionId');

taxDetailsResponse.setTaxId(String.valueOf(getRandomInteger(0,2323233)));
                taxDetailsResponse.setSerCode('serCode');
                taxDetailsResponse.setTaxAuthorityTypeId('taxAuthorityTypeId');
                if(request.DocumentCode == 'SetsNullForResponseWithoutException'){
                    taxDetailsResponse.setImposition(null);
                }else{
                    commercetax.ImpositionResponse imposition = new
commercetax.ImpositionResponse();
                    imposition.setSubType('subtype');
                    imposition.setType('type');
                    taxDetailsResponse.setImposition(imposition);

                }

                if(request.DocumentCode == 'SetsNullForResponseWithoutException'){
                    taxDetailsResponse.setJurisdiction(null);
                }else{
                    commercetax.JurisdictionResponse jurisdiction = new
commercetax.JurisdictionResponse();
                    jurisdiction.setCountry('country');
                    jurisdiction.setRegion('region');
                    jurisdiction.setName('name');
                    jurisdiction.setStateAssignedNumber('stateAssignedNo');
                    jurisdiction.setId('id');
                    jurisdiction.setLevel('level');
                    taxDetailsResponse.setJurisdiction(jurisdiction);
                }
```

```
                taxDetailsResponses.add(taxDetailsResponse);

                lineItemResponse.setTaxes(taxDetailsResponses);
                totalTax +=totalLineTax;
                totalAmount+=lineItem.amount;

                if(request.DocumentCode == 'SetsNullForResponseWithException'){
                    lineItemResponse.setAmountDetails(null);
                }else{
                    commercetax.AmountDetailsResponse amountResponse = new
commercetax.AmountDetailsResponse();
                    amountResponse.setTotalAmountWithTax(totalTax+totalAmount);
                    amountResponse.setExemptAmount(0);
                    amountResponse.setTotalAmount(totalAmount);
                    amountResponse.setTaxAmount(totalTax);
                    lineItemResponse.setAmountDetails(amountResponse);
                }
                lineItemResponse.setEffectiveDate(system.now());
                lineItemResponse.setTaxCode(lineItem.taxCode);
                lineItemResponse.setProductCode(lineItem.ProductCode);
                lineItemResponse.setLineNumber(lineItem.linenumber);
                lineItemResponse.setIsTaxable(true);
                lineItemResponse.setQuantity(lineItem.quantity);
                if(request.DocumentCode == 'SetsNullForResponseWithoutException'){
                    lineItemResponse.setAddresses(null);
                }else{
                    lineItemResponse.setAddresses(addressesRes);
                }
                lineItemResponses.add(lineItemResponse);
            }
            if(request.DocumentCode == 'SetsNullForResponseWithException'){
                lineItemResponses.add(null);
            }
            if(request.documentCode == 'nolines') {
                // logic to skip adding lines to response
            } else {
                    response.setLineItems(lineItemResponses);
            }
            if(request.DocumentCode == 'SetsNullForResponseWithException'){
                response.setAmountDetails(null);
            }else{
                commercetax.AmountDetailsResponse headerAmountResponse = new
commercetax.AmountDetailsResponse();
                headerAmountResponse.setTotalAmountWithTax(totalTax+totalAmount);
                headerAmountResponse.setExemptAmount(0);
                headerAmountResponse.setTotalAmount(totalAmount);
                headerAmountResponse.setTaxAmount(totalTax);
                response.setAmountDetails(headerAmountResponse);
            }
            commercetax.AddressesResponse addressesRes = new
commercetax.AddressesResponse();
           commercetax.AddressResponse addRes = new commercetax.AddressResponse();
```

```
            addRes.setLocationCode('locationCode');
            addressesRes.setShipFrom(addRes);
            addRes.setLocationCode('locationCode');
            addressesRes.setShipTO(addRes);
            addRes.setLocationCode('locationCode');
            addressesRes.setSoldTo(addRes);
            response.setAddresses(addressesRes);
            return response;

        }
        else
            return null;
    }

    public static Integer getRandomInteger(Integer min, Integer max)
    {
        return min + (Integer.valueOf(Math.random()) * (max - min));
    }

    // Method to get the tax rates for the line items based on the address country
 codes
    private Map<String, Double> getTaxes(List<commercetax.TaxLineItemRequest>
lineItems) {
        Set<String> countryCodes = new Set<String>();
        // Collecting all unique the country codes
        for(commercetax.TaxLineItemRequest lineItem : lineItems){
            String country = getCountryFromLineItem(lineItem);
            if (country != null) {
                countryCodes.add(country);
            }
        }
        // Query tax rates for the unique country codes and store it in a map
        Map<String, Double> countryTaxRateMap = new Map<String, Double>();
        if (!countryCodes.isEmpty()) {
            // Query the tax rate from the custom object based on the country code
           List<CountryTaxRate__c> taxRates = [SELECT Country_Code__c, Tax_Rate__c
 FROM CountryTaxRate__c WHERE Country_Code__c IN :countryCodes];
            for (CountryTaxRate__c taxRate : taxRates) {
              countryTaxRateMap.put(taxRate.Country_Code__c, taxRate.Tax_Rate__c);

            }
        }
        return countryTaxRateMap;
    }
    // Method to retrieve the country for the line item
    private String getCountryFromLineItem(commercetax.TaxLineItemRequest lineItem)
 {
        if (lineItem.addresses != null) {
          commercetax.TaxAddressRequest addressRequest = lineItem.addresses.shipTo;

            if (addressRequest != null && addressRequest.country != null) {
                return addressRequest.country;
            }
        }
```

```
        return null;
    }
}
```

- Select `StandardTaxAdapter` as the provider when you create a tax engine record in your org.

Billing Metadata API Types

Metadata API enables you to access some types and feature settings that you can customize in the user interface.

BillingSettings
Represents the settings for Salesforce Billing.

Flow for Billing
Represents the metadata associated with a flow. With Flow, you can create an application that navigates users through a series of screens to query and update records in the database. You can also execute logic and provide branching capability based on user input to build dynamic applications.

SEE ALSO:

Metadata API Developer Guide: Understanding Metadata API

BillingSettings

Represents the settings for Salesforce Billing.

Parent Type and Manifest Access

This type extends the Metadata metadata type and inherits its `fullName` field.

In the package manifest, all the settings metadata types for the org are accessed using the "Settings" name. See Settings for more details.

File Suffix and Directory Location

The `BillingSettings` values are stored in the `BillingSettings.settings` file in the `settings` folder. The `.settings` files are different from other named components, because there's only one settings file for each settings component.

Version

BillingSettings components are available in API version 62.0 and later.

Special Access Rules

These settings are available when Billing is enabled.

Fields

Field Name	Description
`enableBillingSetup`	**Field Type** boolean **Description** Indicates whether to enable Billing setting (`true`) or not (`false`). The default value is `false`.
`enableCrMemoApplicationToPostedInvoices`	**Field Type** boolean **Description** Indicates whether to enable Apply Credits to Posted Invoices setting (`true`) or not (`false`). The default value is `false`.
`enableInvoiceEmailDelivery`	**Field Type** boolean **Description** Indicates whether to enable Configure Email Delivery Settings (`true`) or not (`false`). The default value is `false`. Available in API version 63.0 and later with Revenue Cloud Billing.
`enableInvoicePdfGeneration`	**Field Type** boolean **Description** Indicates whether to enable Document Generation setting (`true`) or not (`false`). The default value is `false`. Available in API version 63.0 and later with Revenue Cloud Billing.
`enableNegInvoiceLnConversionToCrMemoLn`	**Field Type** boolean **Description** Indicates whether to enable Convert Negative Invoice Lines to Credit Memo Lines setting (`true`) or not (`false`). The default value is `false`.
`enableTransactionsApplicationToInvoices`	**Field Type** boolean **Description** Indicates whether to enable Credit Application Level setting (`true`) or not (`false`). The default value is `false`.
`enableTransactionJournalCreation`	**Field Type** boolean

Field Name	Description
	Description Indicates whether to create Transaction Journal records based on the defined general ledger account assignment rules for the billing entities when billing transaction records are created or updated (`true`) or not (`false`). The default value is `false`. Available in API version 63.0 and later with Revenue Cloud Billing. Billing transaction records include these transaction types. • Invoice • Invoice Line • Invoice Line Tax • Credit Memo • Credit Memo Line • Credit Memo Line Tax • Payment • Refund • Payment Line Invoice • Payment Line Invoice Line • Credit Memo Inv Application • Credit Memo Line Invoice Line
`enableTrxnAmountsStorageInCorpCurrency`	**Field Type** boolean **Description** Indicates whether to allow conversion of amounts of the Invoice, Invoice Line, Credit Memo, and Credit Memo Line records to your corporate currency (`true`) or not (`false`). The default value is `false`. Available in API version 63.0 and later. Store the converted amounts in corporate currency-specific amount fields.

Declarative Metadata Sample Definition

The following is an example of a BillingSettings component.

```
<BillingSettings xmlns="http://soap.sforce.com/2006/04/metadata">
    <enableBillingSetup>true</enableBillingSetup>
    <enableInvoicePdfGeneration>true</enableInvoicePdfGeneration>
   <enableTransactionsApplicationToInvoices>true</enableTransactionsApplicationToInvoices>

   <enableCrMemoApplicationToPostedInvoices>true</enableCrMemoApplicationToPostedInvoices>

    <enableInvoiceEmailDelivery>true</enableInvoiceEmailDelivery>
    <enableTransactionJournalCreation>true</enableTransactionJournalCreation>
    <enableTrxnAmountsStorageInCorpCurrency>true</enableTrxnAmountsStorageInCorpCurrency>
</BillingSettings>
```

The following is an example `package.xml` that references the previous definition.

```
<?xml version="1.0" encoding="UTF-8"?>
<Package xmlns="http://soap.sforce.com/2006/04/metadata">
    <types>
        <members>Billing</members>
        <name>Settings</name>
    </types>
    <version>63.0</version>
</Package>
```

Wildcard Support in the Manifest File

The wildcard character `*` (asterisk) in the `package.xml` manifest file doesn't apply to metadata types for feature settings. The wildcard applies only when retrieving all settings, not for an individual setting. For details, see Settings. For information about using the manifest file, see Deploying and Retrieving Metadata with the Zip File.

Flow for Billing

Represents the metadata associated with a flow. With Flow, you can create an application that navigates users through a series of screens to query and update records in the database. You can also execute logic and provide branching capability based on user input to build dynamic applications.

FlowActionCall

Billing exposes additional actionType values for the FlowActionCall Metadata type.

Field Name	Field Type	Description
`actionType`	InvocableActionType (enumeration of type string)	Required. The action type. Additional valid values only for Billing include: • `applyCredit`—Apply a credit memo or credit memo line to an invoice or invoice line, respectively. • `unapplyCredit`—Unapply a credit memo or credit memo line from an invoice or invoice line, respectively. • `postDraftInvoice`—Update the status of an invoice from `Draft` to `Posted` for a credit memo application. • `postDraftInvoiceBatchRun`—Update the status of a batch of invoices from `Draft` to `Posted` for a credit memo application. • `createBillingSchedulesFromBillingTransaction`—Create one or more billing schedules for a specified billing transaction ID. • `recoverBillingSchedules`—Recover one or more billing schedules in the `Error` or `Processing` status. • `generateInvoiceDocuments`—Asynchronously generate PDF documents for the invoices associated with an invoice batch run record that are in the `Draft` or `Posted` status.

CHAPTER 10 Revenue Cloud Associated Objects

In this chapter ...

- StandardObjectNameChangeEvent
- StandardObjectNameFeed
- StandardObjectNameHistory
- StandardObjectNameOwnerSharingRule
- StandardObjectNameShare
- Event
- Task

This section provides a list of objects associated to standard objects of Revenue Cloud with their standard fields.

Some fields may not be listed for some objects. To see the system fields for each object, see System Fields in the *Object Reference for Salesforce and Lightning Platform*.

To verify the complete list of fields for an object, use a describe call from the API or inspect with an appropriate tool. For example, inspect the WSDL or use a schema viewer.

*StandardObjectName*ChangeEvent

A ChangeEvent object is available for each object that supports Change Data Capture. You can subscribe to a stream of change events using Change Data Capture to receive data tied to record changes in Salesforce. Changes include record creation, updates to an existing record, deletion of a record, and undeletion of a record. A change event isn't a Salesforce object—it doesn't support CRUD operations or queries. It's included in the object reference so you can discover which Salesforce objects support change events.

Supported Calls

`describeSObjects()`

Special Access Rules

- All objects may not be available in your org. Some objects require specific feature settings and permissions to be enabled.
- For more special access rules, if any, see the documentation for the standard object. For example, for AccountChangeEvent, see the special access rules for Account.

Change Event Support

Change events are available for all custom objects and a subset of standard objects. Change events that correspond to custom settings are partially supported. They aren't supported in Apex triggers but are supported in other types of subscribers. For more information about standard object support, see the Objects That Support Change Events section below.

Change Event Name

The name of a change event is based on the name of the corresponding object for which it captures the changes.

Standard Object Change Event Name

```
<Standard_Object_Name>ChangeEvent
```

Example: `AccountChangeEvent`

Custom Object Change Event Name

```
<Custom_Object_Name>__ChangeEvent
```

Example: `MyCustomObject__ChangeEvent`

Change Event Fields

The fields that a change event can include correspond to the fields on the associated parent Salesforce object, with a few exceptions. For example, AccountChangeEvent fields correspond to the fields on Account.

The fields that a change event doesn't include are:

- The `IsDeleted` system field.
- The `SystemModStamp` system field.
- Any field whose value isn't on the record and is derived from another record or from a formula, except roll-up summary fields, which are included. Examples are formula fields. Examples of fields with derived values include `LastActivityDate` and `PhotoUrl`.

Each change event also contains header fields. The header fields are included inside the `ChangeEventHeader` field. They contain information about the event, such as whether the change was an update or delete and the name of the object, like Account.

In addition to the event payload, the event schema ID is included in the `schema` field. Also included is the event-specific field, `replayId`, which is used for retrieving past events.

Event Message Example

This example is an event message in JSON format for a new account record creation.

```
{
  "schema": "IeRuaY6cbI_HsV8Rv1Mc5g",
  "payload": {
    "ChangeEventHeader": {
      "entityName": "Account",
      "recordIds": [
        "<record_ID>"
      ],
      "changeType": "CREATE",
      "changeOrigin": "com/salesforce/api/soap/51.0;client=SfdcInternalAPI/",
      "transactionKey": "0002343d-9d90-e395-ed20-cf416ba652ad",
      "sequenceNumber": 1,
      "commitTimestamp": 1612912679000,
      "commitNumber": 10716283339728,
      "commitUser": "<User_ID>"
    },
    "Name": "Acme",
    "Description": "Everyone is talking about the cloud. But what does it mean?",
    "OwnerId": "<Owner_ID>",
    "CreatedDate": "2021-02-09T23:17:59Z",
    "CreatedById": "<User_ID>",
    "LastModifiedDate": "2021-02-09T23:17:59Z",
    "LastModifiedById": "<User_ID>"
  },
  "event": {
    "replayId": 6
  }
}
```

API Version and Schema

When you subscribe to change events, the subscription uses the latest API version and the event messages received reflect the latest field definitions. For more information, see API Version and Event Schema in the *Change Data Capture Developer Guide*.

Usage

For more information about Change Data Capture, see Change Data Capture Developer Guide.

*StandardObjectName*Feed

*StandardObjectName*Feed is the model for all feed objects associated with standard objects. These objects represent the posts and feed-tracked changes of a standard object.

The object name is variable and uses *StandardObjectName*Feed syntax. For example, AccountFeed represents the posts and feed-tracked changes on an account record. We list the available associated feed objects at the end of this topic. For specific version information, see the documentation for the standard object.

Supported Calls

`delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve()`

Special Access Rules

In the internal org, users can delete all feed items they created. This rule varies in Experience Cloud sites where threaded discussions and delete-blocking are enabled. Site members can delete all feed items they created, provided the feed items don't have content nested under them—like a comment, answer, or reply. Where the feed item has nested content, only feed moderators and users with the Modify All Data permission can delete threads.

To delete feed items they didn't create, users must have one of these permissions:

- Modify All Data
- Modify All Records on the parent object, like Account for AccountFeed
- Moderate Chatter

  Note: Users with the Moderate Chatter permission can delete only the feed items and comments they can see.

  Only users with this permission can delete items in unlisted groups.

For more special access rules, if any, see the documentation for the standard object. For example, for AccountFeed, see the special access rules for Account.

Fields

Field	Details
`BestCommentId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The ID of the comment marked as best answer on a question post. This field is available in API version 44.0 and later.
`Body`	**Type** textarea

Field	Details
	Properties Nillable, Sort **Description** The body of the post. Required when `Type` is `TextPost`. Optional when `Type` is `ContentPost` or `LinkPost`.
`CommentCount`	**Type** int **Properties** Filter, Group, Sort **Description** The number of comments associated with this feed item. In a feed that supports pre-moderation, `CommentCount` isn't updated until a comment is published. For example, say that you comment on a post that already has one published comment and your comment triggers moderation. Now there are two comments on the post, but the count says there's only one. In a moderated feed, comments aren't counted until approved by an admin or someone with Can Approve Feed Post and Comment or Modify All Data. Feed moderation has implications on how you retrieve feed comments. In a moderated feed, rather than retrieving comments by looping through `CommentCount`, go through pagination until the end of comments is returned.
`ConnectionId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** When a PartnerNetworkConnection modifies a record that is tracked, the `CreatedBy` field contains the ID of the system administrator. The `ConnectionId` contains the ID of the PartnerNetworkConnection. Available if Salesforce to Salesforce is enabled for your organization.
`ContentData`	**Type** base64 **Properties** Nillable **Description** Available in API version 36.0 and earlier only. Required if `Type` is `ContentPost`. Encoded file data in any format, and can't be 0 bytes. Setting this field automatically sets `Type` to `ContentPost`.
`ContentDescription`	**Type** textarea

Field	Details
	Properties Nillable, Sort **Description** Available in API version 36.0 and earlier only. The description of the file specified in `ContentData`.
`ContentFileName`	**Type** string **Properties** Group, Nillable, Sort **Description** Available in API version 36.0 and earlier only. This field is required if `Type` is `ContentPost`.The name of the file uploaded to the feed. Setting `ContentFileName` automatically sets `Type` to `ContentPost`.
`ContentSize`	**Type** int **Properties** Group, Nillable, Sort **Description** Available in API version 36.0 and earlier only. The size of the file (in bytes) uploaded to the feed. This field is read-only and is automatically determined during insert.
`ContentType`	**Type** string **Properties** Group, Nillable, Sort **Description** Available in API version 36.0 and earlier only. The MIME type of the file uploaded to the feed. This field is read-only and is automatically determined during insert.
`FeedPostId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** This field was removed in API version 22.0, and is available in earlier versions for backward compatibility only. ID of the associated FeedPost. A FeedPost represents the following types of changes in a feed item: changes to tracked fields, text posts, link posts, and content posts.

Field	Details
InsertedById	**Type** reference **Properties** Group, Nillable, Sort **Description** ID of the user who added this item to the feed. For example, if an application migrates posts and comments from another application into a feed, the `InsertedBy` value is set to the ID of the context user.
isRichText	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Indicates whether the feed item `Body` contains rich text. If you post a rich text feed comment using SOAP API, set `IsRichText` to `true` and escape HTML entities from the body. Otherwise, the post is rendered as plain text. Rich text supports the following HTML tags: • `<p>`Though the ` ` tag isn't supported, you can use `<p> </p>` to create lines. • `<a>` • `<b>` • `<code>` • `<i>` • `<u>` • `<s>` • `<ul>` • `<ol>` • `<li>` • `<img>` The `<img>` tag is accessible only through the API and must reference files in Salesforce similar to this example: `<img src="sfdc://069B0000000omjh"></img>` In API version 35.0 and later, the system replaces special characters in rich text with escaped HTML. In API version 34.0 and prior, all rich text appears as a plain-text representation.
LikeCount	**Type** int **Properties** Filter, Group, Sort

Field	Details
	Description The number of likes associated with this feed item.
`LinkUrl`	**Type** url **Properties** Nillable, Sort **Description** The URL of a `LinkPost`.
`NetworkScope`	**Type** picklist **Properties** Group, Nillable, Restricted picklist, Sort **Description** Specifies whether this feed item is available in the default Experience Cloud site, a specific Experience Cloud site, or all sites. This field is available in API version 26.0 and later, if digital experiences is enabled for your org. `NetworkScope` can have the following values: • `NetworkId`—The ID of the Experience Cloud site in which the FeedItem is available. If left empty, the feed item is only available in the default Experience Cloud site. • `AllNetworks`—The feed item is available in all Experience Cloud sites. Note the following exceptions for `NetworkScope`: • Only feed items with a Group or User parent can set a `NetworkId` or a null value for `NetworkScope`. • For feed items with a record parent, users can set `NetworkScope` only to `AllNetworks`. • You can't filter a feed item on the `NetworkScope` field.
`ParentId`	**Type** reference **Properties** Filter, Group, Sort **Description** ID of the record that is tracked in the feed. The detail page for the record displays the feed.
`RelatedRecordId`	**Type** reference **Properties** Group, Nillable, Sort

Field	Details
	Description ID of the ContentVersion record associated with a `ContentPost`. This field is null for all posts except `ContentPost`.
`Title`	**Type** string **Properties** Group, Nillable, Sort **Description** The title of the feed item. When the `Type` is `LinkPost`, the `LinkUrl` is the URL and this field is the link name.
`Type`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** The type of feed item: • `ActivityEvent`—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging). For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence. • `AdvancedTextPost`—created when a user posts a group announcement and, in Lightning Experience as of API version 39.0 and later, when a user shares a post. • `AnnouncementPost`—Not used. • `ApprovalPost`—generated when a user submits an approval. • `BasicTemplateFeedItem`—Not used. • `CanvasPost`—a post made by a canvas app posted on a feed. • `CollaborationGroupCreated`—generated when a user creates a public group. • `CollaborationGroupUnarchived`—Not used. • `ContentPost`—a post with an attached file. • `CreatedRecordEvent`—generated when a user creates a record from the publisher. • `DashboardComponentAlert`—generated when a dashboard metric or gauge exceeds a user-defined threshold. • `DashboardComponentSnapshot`—created when a user posts a dashboard snapshot on a feed. • `LinkPost`—a post with an attached URL. • `PollPost`—a poll posted on a feed.

Field	Details
	• `ProfileSkillPost`—generated when a skill is added to a user's Chatter profile. • `QuestionPost`—generated when a user posts a question. • `ReplyPost`—generated when Chatter Answers posts a reply. • `RypplePost`—generated when a user creates a Thanks badge in WDC. • `TextPost`—a direct text entry on a feed. • `TrackedChange`—a change or group of changes to a tracked field. • `UserStatus`—automatically generated when a user adds a post. Deprecated. The following values appear in the `Type` picklist for all feed objects but apply only to CaseFeed: • `CaseCommentPost`—generated event when a user adds a case comment for a case object • `EmailMessageEvent`—generated event when an email related to a case object is sent or received • `CallLogPost`—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event. • `ChangeStatusPost`—generated event when a user changes the status of a case • `AttachArticleEvent`—generated event when a user attaches an article to a case If you set `Type` to `ContentPost`, also specify `ContentData` and `ContentFileName`.
`Visibility`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** Specifies whether this feed item is available to all users or internal users only. This field is available in API version 26.0 and later, if digitial experiences is enabled for your organization. `Visibility` can have the following values: • `AllUsers`—The feed item is available to all users who have permission to see the feed item. • `InternalUsers`—The feed item is available to internal users only. Note the following exceptions for `Visibility`: • For record posts, `Visibility` is set to `InternalUsers` for all internal users by default. • External users can set `Visibility` only to `AllUsers`. • On user and group posts, only internal users can set `Visibility` to `InternalUsers`.

Usage

A feed for an object is automatically created when a user enables feed tracking for the object. Use feeds to track changes to records. For example, `AccountFeed` tracks changes to an account record. Use feed objects to retrieve the content of feed fields, such as type of feed or feed ID.

- `NewsFeed` and `UserProfileFeed` are available in API version 18.0 through API version 26.0. In API version 27.0 and later, `NewsFeed` and `UserProfileFeed` are no longer available in SOAP API. Use Connect REST API to access `NewsFeed` and `UserProfileFeed`.

  Use the `NewsFeed` object to query and retrieve lead feed items associated with a converted lead record.

- For `NewsFeed` and `UserProfileFeed`, users who don't have the View All Data permission have the following limitations when querying records: Must specify a `LIMIT` clause and the limit must be less than or equal to 1000. Can include a `WHERE` clause that references object fields, but can't include references to fields in related objects. For example, you can filter by `CreatedDate` or `ParentId`, but not by `Parent.Name`. Can include an `ORDER BY` clause that references object fields, but can't include references to fields in related objects. For example, `ORDER BY CreatedDate` or `ParentId`, but not by `Parent.Name`. To query for the most recent feed items, `ORDER BY CreatedDate DESC, Id DESC`.

  Note the following SOQL restrictions. No SOQL limit if logged-in user has View All Data permission. If not, specify a `LIMIT` clause of 1,000 records or fewer. SOQL `ORDER BY` on fields using relationships isn't available. Use `ORDER BY` on fields on the root object in the SOQL query.

- The name *`Article Type`*`__`Feed is variable, where *Article Type* is the object name for the article type associated with the article. For example, `Offer__Feed` represents a feed on an article of type *Offer*.
- Field Service must be enabled in your organization for `ServiceAppointmentFeed`, `ServiceCrewFeed`, `ServiceMemberFeed`, `ServiceResourceCapacityFeed`, `ServiceResourceFeed`, `ServiceResourceSkillFeed`, `ServiceTerritoryFeed`, `ServiceTerritoryMemberFeed`, and `SkillRequirementFeed`.
- For `WorkOrderFeed`, Work Orders or Field Service must be enabled in your organization.
- On `UserFeed`, if you use the `FeedComment` object to comment on a user record, the user can delete the comment. For example, if John Smith adds a comment to the feed on Sasha Jones' user record, Sasha can delete the comment.

*`StandardObjectName`*History

*`StandardObjectName`*History is the model for all history objects associated with standard objects. These objects represent the history of changes to the values in the fields of a standard object.

The object name is variable and uses *`StandardObjectName`*History syntax. For example, AccountHistory represents the history of changes to the values of an account record's fields. We list the available associated history objects at the end of this topic. For specific version information, see the documentation for the standard object.

Supported Calls

`describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

You can also enable `delete()` in API version 42.0 and later. See Enable delete of Field History and Field History Archive.

Special Access Rules

For specific special access rules, if any, see the documentation for the standard object. For example, for AccountHistory, see the special access rules for Account.

Fields

Field Name	Details
*StandardObjectName*Id	**Type** reference **Properties** Filter, Group, Sort **Description** ID of the standard object.
DataType	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** Data type of the field that was changed.
Field	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** Name of the field that was changed.
NewValue	**Type** anyType **Properties** Nillable, Sort **Description** New value of the field that was changed.
OldValue	**Type** anyType **Properties** Nillable, Sort **Description** Old value of the field that was changed.

*StandardObjectName*OwnerSharingRule

*StandardObjectName*OwnerSharingRule is the model for all owner sharing rule objects associated with standard objects. These objects represent a rule for sharing a standard object with users other than the owner.

The object name is variable and uses *StandardObjectName*OwnerSharingRule syntax. For example, ChannelProgramOwnerSharingRule is a rule for sharing a channel program with users other than the channel program owner. The available associated owner sharing rule objects are listed at the end of this topic. For specific version information, see the standard object documentation.

Note: To enable access to this object, contact Salesforce customer support. But we recommend that you use Metadata API to programmatically update owner sharing rules instead because it triggers automatic sharing rule recalculation. The SharingRules Metadata API type is enabled for all orgs.

Supported Calls

`create(), delete(), describeSObjects(), getDeleted(), getUpdated(), query(), retrieve(), update(), upsert()`

Special Access Rules

For specific special access rules, if any, see the documentation for the standard object. For example, for ChannelProgramOwnerSharingRule, see the special access rules for ChannelProgram.

Fields

Field Name	Details
`AccessLevel`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** Determines the level of access users have to records. Values are: • `Read` (read only) • `Edit` (read/write)
`Description`	**Type** textarea **Properties** Create, Filter, Nillable, Sort, Update **Description** Description of the sharing rule. Maximum length is 1,000 characters.

Field Name	Details
`DeveloperName`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The unique name of the object in the API. This name can contain only underscores and alphanumeric characters and must be unique in your org. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package, and the changes are reflected in a subscriber's organization. Note: When creating large sets of data, always specify a unique `DeveloperName` for each record. If no `DeveloperName` is specified, performance can slow while Salesforce generates one for each record.
`GroupId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** ID of the source group. Records that are owned by users in the source group trigger the rule to give access.
`Name`	**Type** string **Properties** Create, Filter, Group, idLookup, Sort, Update **Description** Label of the sharing rule as it appears in the UI. Maximum length is 80 characters.
`UserOrGroupId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** ID of the user or group that you're granting access to.

*StandardObjectName*Share

*StandardObjectName*Share is the model for all share objects associated with standard objects. These objects represent a sharing entry on the standard object.

The object name is variable and uses *StandardObjectName*Share syntax. For example, AccountBrandShare is a sharing entry on an account brand. For specific version information, see the standard object documentation.

You can only create, edit, and delete sharing entries for standard objects whose `RowCause` field is set to `Manual`. Sharing entries for standard objects with different `RowCause` values are created as a result of your Salesforce org's sharing configuration and are read-only. For some sharing mechanisms, such as sharing sets, sharing entries aren't stored at all.

Note: While Salesforce currently maintains read-only sharing entries for multiple sharing mechanisms, it's possible that we'll stop storing certain share records to improve performance. As a best practice, don't create customizations that rely on the availability of these sharing entries. Any changes to sharing behavior will be communicated before they occur.

Supported Calls

`create()`, `delete()`, `describeSObjects()`, `query()`, `retrieve()`, `update()`, `upsert()`

Special Access Rules

For specific special access rules, if any, see the documentation for the standard object. For example, for AccountBrandShare, see the special access rules for AccountBrand.

Fields

Field Name	Details
`AccessLevel`	**Type** picklist **Properties** Create, Filter, Group, Restricted picklist, Sort, Update **Description** The level of access allowed. Values are: • `All` (owner) • `Edit` (read/write) • `Read` (read only)
`ParentId`	**Type** reference **Properties** Create, Filter, Group, Sort **Description** ID of the parent record.

Field Name	Details
RowCause	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort **Description** Reason that the sharing entry exists.
UserOrGroupId	**Type** reference **Properties** Create, Filter, Group, Sort **Description** ID of the user or group that has been given access to the object.

Event

Represents an event in the calendar. In the user interface, event and task records are collectively referred to as activities.

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.

Note:

- An EventRelation object can't be related to a child event, and child events don't include the invitee related list.
- `query()`, `delete()`, and `update()` aren't allowed with events related to more than one contact in API versions 25.0 and earlier.
- `create()` and `update()` aren't available for read-only fields on Lightning Experience event series.
- `upsert()` and `undelete()` aren't supported for syncing changes made to events through the API using the feature Lightning Sync.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Details
AcceptedEventInviteeIds	**Type** JunctionIdList

Field	Details
	Properties Create, Update **Description** A string array of contact or lead IDs who accepted this event. This `JunctionIdList` is linked to the `AcceptedEventRelation` child relationship. Warning: Adding a `JunctionIdList` field name to the `fieldsToNull` property deletes all related junction records. This action can't be undone.
`AccountId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** Represents the ID of the related account. The `AccountId` is determined as follows. If the value of `WhatId` is any of the following objects, then Salesforce uses that object's `AccountId`. • Account • Opportunity • Contract • Custom object that's a child of Account If the value of the `WhatId` field is any other object, and the value of the `WhoId` field is a contact object, then Salesforce uses that contact's `AccountId`. If your org uses Shared Activities, Salesforce uses the `AccountId` of the primary contact. Otherwise, Salesforce sets the value of the `AccountId` field to `null`. For information on IDs, see ID Field Type. This is a relationship field. **Relationship Name** Account **Relationship Type** Lookup **Refers To** Account
`ActivityDate`	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Contains the event's due date if the `IsAllDayEvent` flag is set to `true`. This field is a date field with a timestamp that's always set to midnight in the Coordinated Universal Time

Field	Details
	(UTC) time zone. Don't attempt to alter the timestamp to account for time zone differences. Label is **Due Date Only**. This field is required in API versions 12.0 and earlier if the `IsAllDayEvent` flag is set to `true`. The value for this field and `StartDateTime` must match, or one of them must be `null`.
`ActivityDateTime`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** Contains the event's due date if the `IsAllDayEvent` flag is set to `false`. The time portion of this field is always transferred in the Coordinated Universal Time (UTC) time zone. Translate the time portion to or from a local time zone for the user or the application, as appropriate. Label is **Due Date Time**. This field is required in API versions 12.0 and earlier if the `IsAllDayEvent` flag is set to `false`. The value for this field and `StartDateTime` must match, or one of them must be `null`.
`ClientGuid`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The client globally unique identifier identifies the external API client used to create the event. Label is **Client GUID**.
`CurrencyIsoCode`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Available only for orgs with the multicurrency feature enabled. Contains the ISO code for any currency allowed by the organization.
`DeclinedEventInviteeIds`	**Type** JunctionIdLIst **Properties** Create, Update **Description** A string array of contact, lead, or user IDs who declined this event. This `JunctionIdList` is linked to the `DeclinedEventRelation` child relationship.

Field	Details
	Warning: Adding a `JunctionIdList` field name to the `fieldsToNull` property deletes all related junction records. This action can't be undone.
`Description`	**Type** textarea **Properties** Create, Nillable, Update **Description** Contains a text description of the event. Limit: 32,000 characters.
`Division`	**Type** picklist **Properties** Defaulted on create, Filter, Group, Restricted picklist, Sort **Description** A logical segment of your organization's data. For example, if your company is organized into different business units, you could create a division for each business unit, such as "North America," "Healthcare," or "Consulting." Available only if the organization has the Division permission enabled.
`DurationInMinutes`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Contains the event length, in minutes. Even though this field represents a temporal value, it's an integer type—not a Date/Time type. Required in API versions 12.0 and earlier if `IsAllDayEvent` is false. In API versions 13.0 and later, this field is optional, depending on the following: • If `IsAllDayEvent` is true, you can supply a value for either `DurationInMinutes` or `EndDateTime`. Supplying values in both fields is allowed if the values add up to the same amount of time. If both fields are `null`, the duration defaults to one day. • If `IsAllDayEvent` is false, a value must be supplied for either `DurationInMinutes` or `EndDateTime`. Supplying values in both fields is allowed if the values add up to the same amount of time. If the multiday event feature is enabled, then API versions 13.0 and later support values greater than 1440 for the `DurationInMinutes` field. API versions 12.0 and earlier can't access event objects whose `DurationInMinutes` is greater than 1440. For more information, see **Multiday Events**. Depending on your API version, errors with the `DurationInMinutes` and `EndDateTime` fields may appear in different places. • Versions 38.0 and before—Errors always appear in the `DurationInMinutes` field.

Field	Details
	• Versions 39.0 and later—If there's no value for the `DurationInMinutes` field, errors appear in the `EndDateTime` field. Otherwise, they appear in the `DurationInMinutes` field.
`EndDate`	**Type** date **Properties** Filter, Group, Nillable, Sort **Description** Read-only. Available in API versions 46.0 and later. This field supplies the date value that appears in the EndDateTime field. This field is a date field with a timestamp that is always set to midnight in the Coordinated Universal Time (UTC) time zone.
`EndDateTime`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** Available in API versions 13.0 and later. The time portion of this field is always transferred in the Coordinated Universal Time (UTC) time zone. Translate the time portion to or from a local time zone for the user or the application, as appropriate. This field is optional, depending on the following: • If `IsAllDayEvent` is true, you can supply a value for either `DurationInMinutes` or `EndDateTime`. Supplying values in both fields is allowed if the values add up to the same amount of time. If both fields are `null`, the duration defaults to one day. • If `IsAllDayEvent` is false, a value must be supplied for either `DurationInMinutes` or `EndDateTime`. Supplying values in both fields is allowed if the values add up to the same amount of time. Depending on your API version, errors with the `DurationInMinutes` and `EndDateTime` fields may appear in different places. • Versions 38.0 and before—Errors always appear in the `DurationInMinutes` field. • Versions 39.0 and later—If there's no value for the `DurationInMinutes` field, errors appear in the `EndDateTime` field. Otherwise, they appear in the `DurationInMinutes` field.
`EventSubtype`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort **Description** Provides standard subtypes to facilitate creating and searching for events. This field isn't updateable.

Field	Details
`EventWhoIds`	**Type** JunctionIdList **Properties** Create, Update **Description** A string array of contact or lead IDs used to create many-to-many relationships with a shared event. `EventWhoIds` is available when the shared activities setting is enabled. The first contact or lead ID in the list becomes the primary `WhoId` if you don't specify a primary `WhoId`. If you set the `EventWhoIds` field to null, all entries in the list are deleted and the value of `WhoId` is added as the first entry. Warning: Adding a `JunctionIdList` field name to the `fieldsToNull` property deletes all related junction records. This action can't be undone.
`GroupEventType`	**Type** picklist **Properties** Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort **Description** Read-only. Available in API versions 19.0 and later. The possible values are: • `0` (Non–group event)—An event with no invitees. • `1` (Group event)—An event with invitees. • `2` (Proposed event)—An event created when a user requests a meeting with a contact, lead, or person account using the Salesforce user interface. When the user confirms the meeting, the proposed event becomes a group event. You can't create, edit, or delete proposed events in the API. This value is no longer used in API version 41.0 and later. • `3` (IsRecurrence2 Series Pattern)—An event representing an event series recurrence pattern in Lightning Experience.
`IsAllDayEvent`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the `ActivityDate` field (`true`) or the `ActivityDateTime` field (`false`) is used to define the date or time of the event. Label is **All-Day Event**. See also `DurationInMinutes` and `EndDateTime`.
`IsArchived`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort

Field	Details
	Description Indicates whether the event has been archived.
`IsChild`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Indicates whether the event is a child of another event (`true`) or not (`false`). For a child event, you can update `IsReminderSet` and `ReminderDateTime` only. You can query and delete a child event. If the objects related to the child event are different from those objects related to the parent event (this difference is possible if you use API version 25.0 or earlier) and one of the objects related to the child event is deleted, the objects related to the parent event are updated to ensure data integrity.
`IsClientManaged`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Indicates whether the event is managed by an external client. If the value of this field is false, the event isn't owned or managed by an external client, and Salesforce can be used to update it. If the value is true, Salesforce can be used to change only noncritical fields on the event. Label is **Is Client Managed**.
`IsGroupEvent`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Indicates whether the event is a group event—that is, whether it has invitees (`true`) or not (`false`).
`IsPrivate`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether users other than the creator of the event can (`false`) or can't (`true`) see the event details when viewing the event user's calendar. However, users with the View All Data or Modify All Data permission can see private events in reports and searches, or when viewing other users' calendars. Private events can't be associated with opportunities, accounts, cases, campaigns, contracts, leads, or contacts. Label is **Private**.

Field	Details
`IsRecurrence`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort **Description** Indicates whether a Salesforce Classic event is scheduled to repeat itself (`true`) or only occurs one time (`false`). This field is read-only when updating records, but not when creating them. If this field value is `true`, then `RecurrenceEndDateOnly`, `RecurrenceStartDateTime`, `RecurrenceType`, and any recurrence fields associated with the given recurrence type must be populated. Label is **Create recurring series of events**.
`IsRecurrence2`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Read-only. This field is available in API version 44.0 and later. Indicates whether a Lightning Experience event is scheduled to repeat (`true`) or only occurs one time (`false`) . If this field value is true, then `Recurrence2PatternText` and `Recurrence2PatternVersion` must be populated. Label is **Repeat**.
`IsRecurrence2Exception`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Read-only. This field is available in API version 44.0 and later. Indicates whether an individual event in a Lightning Experience event series has a recurrence pattern that's different from the rest of the series, making it an exception.
`IsRecurrence2Exclusion`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Read-only. This field is available in API version 44.0 and later. Indicates when updates to a Lightning Experience event series recurrence pattern have been made, but affect future event occurrences only. For past event occurrences, `IsRecurrence2Exclusion` is set to `true`, excluding past occurrences from the series recurrence pattern.
`IsReminderSet`	**Type** boolean

Field	Details
	Properties Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether the activity is a reminder (`true`) or not (`false`).
`IsVisibleInSelfService`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether an event associated with an object can be viewed in the Customer Portal (`true`) or not (`false`). If your org has enabled digital experiences, events marked `IsVisibleInSelfService` are visible to any external user in the Experience Cloud site, as long as the user has access to the record the event was created on. This field is available when • Customer Portal or partner portal is enabled OR • Digital experiences is enabled and you have Customer Portal or partner portal licenses
`Location`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Contains the location of the event.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Contains the ID of the user or public calendar who owns the event. Label is **Assigned to ID**. This is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Calendar, User

Field	Details
`Recurrence2PatternStartDate`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** Read-only. This field is available in API version 44.0 and later. Indicates the date and time when the Lightning Experience event series begins. The time portion of this field is always transferred in the Coordinated Universal Time (UTC) time zone. Translate the time portion to or from a local time zone for the user or the application, as appropriate.
`Recurrence2PatternText`	**Type** textarea **Properties** Create, Nillable **Description** The RRULE that describes the recurrence pattern for Lightning Experience event series. Supports a subset of the RFC 5545 standard for internet calendaring and scheduling. See the Event Series section in this topic for usage examples. This field has a maximum length of 512 characters. This field is available in API version 44.0 and later, and has the `Create` property in API version 52.0 and later.
`Recurrence2PatternTimeZone`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** This field is available in API version 44.0 and later. Indicates the time zone in which the Lightning Experience event series was created or updated. This field uses standard Java TimeZone IDs. For example, America/Los_Angeles.
`Recurrence2PatternVersion`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort, **Description** Read-only. This field is available in API version 44.0 and later. Indicates the standard specifications for Lightning Experience event series recurrence patterns. The only possible value is `1` (RFC 5545 v4 RRULE)—RFC 5545 is a standard set of specifications for internet calendaring and scheduling that `IsRecurrence2` adheres to for series recurrence patterns. RFC 5545 specifications for series recurrence patterns are called RRULES. For examples of RRULE usage, see the Lightning Experience Event Series and Recurring Events section in this topic.

Field	Details
`RecurrenceActivityId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** Read-only. Not required on create. Contains the ID of the main record of the Salesforce Classic recurring event. Subsequent occurrences have the same value in this field.
`RecurrenceDayOfMonth`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Indicates the day of the month on which the event repeats.
`RecurrenceDayOfWeekMask`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Indicates the day or days of the week on which the Salesforce Classic recurring event repeats. This field contains a bitmask. The values are as follows: • Sunday = `1` • Monday = `2` • Tuesday = `4` • Wednesday = `8` • Thursday = `16` • Friday = `32` • Saturday = `64` Multiple days are represented as the sum of their numerical values. For example, Tuesday and Thursday = 4 + 16 = 20.
`RecurrenceEndDateOnly`	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Indicates the last date on which the event repeats. For multiday Salesforce Classic recurring events, this date is the day on which the last occurrence starts. This field is a date field with a timestamp that is always set to midnight in the Coordinated Universal Time (UTC) time zone. Don't attempt to alter the timestamp to account for time zone differences.

Field	Details
`RecurrenceInstance`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Indicates the frequency of the Salesforce Classic event's recurrence. For example, `2nd` or `3rd`.
`RecurrenceInterval`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Indicates the interval between Salesforce Classic recurring events.
`RecurrenceMonthOfYear`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Indicates the month in which the Salesforce Classic recurring event repeats.
`RecurrenceStartDateTime`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** Indicates the date and time when the Salesforce Classic recurring event begins. The value must precede the `RecurrenceEndDateOnly`. The time portion of this field is always transferred in the Coordinated Universal Time (UTC) time zone. Translate the time portion to or from a local time zone for the user or the application, as appropriate.
`RecurrenceTimeZoneSidKey`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Indicates the time zone associated with a Salesforce Classic recurring event. For example, "UTC-8:00" for Pacific Standard Time.
`RecurrenceType`	**Type** picklist

Field	Details
	Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Indicates how often the Salesforce Classic event repeats. For example, daily, weekly, or every nth month (where "nth" is defined in `RecurrenceInstance`).
`ReminderDateTime`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** Represents the time when the reminder is scheduled to fire, if `IsReminderSet` is set to `true`. If `IsReminderSet` is set to `false`, then the user may have deselected the reminder checkbox in the Salesforce user interface, or the reminder has already fired at the time indicated by the value.
`ShowAs`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Indicates how this event appears when another user views the calendar: Busy, Out of Office, or Free. Label is **Show Time As**.
`StartDateTime`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** Indicates the start date and time of the event. Available in versions 13.0 and later. If the Event `IsAllDayEvent` flag is set to true (indicating that it's an all-day Event), then the event start date information is contained in the `StartDateTime` field. The time portion of this field is always transferred in the Coordinated Universal Time (UTC) time zone. Translate the time portion to or from a local time zone for the user or the application, as appropriate. If the Event `IsAllDayEvent` flag is set to false (indicating that it isn't an all-day event), then the event start date information is contained in the `StartDateTime` field. The time portion is always transferred in the Coordinated Universal Time (UTC) time zone. You need to translate the time portion to or from a local time zone for the user or the application, as appropriate. If this field has a value, then `ActivityDate` and `ActivityDateTime` must either be `null` or match the value of this field.

Field	Details
`Subject`	**Type** combobox **Properties** Create, Filter, Nillable, Sort, Update **Description** The subject line of the event, such as Call, Email, or Meeting. Limit: 255 characters.
`Type`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Indicates the event type, such as Call, Email, or Meeting.
`UndecidedEventInviteeIds`	**Type** JunctionIdList **Properties** Create, Update **Description** A string array of contact, lead, or user IDs who are undecided about this event. This `JunctionIdList` is linked to the `UndecidedEventRelation` child relationship. Warning: Adding a `JunctionIdList` field name to the `fieldsToNull` property deletes all related junction records. This action can't be undone.
`WhatCount`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** Available if your organization has enabled Shared Activities. Represents the count of related EventRelations pertaining to the `WhatId`. The count of the `WhatId` must be *1* or less.
`WhatId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The `WhatId` represents nonhuman objects such as accounts, opportunities, campaigns, cases, or custom objects. `WhatId`s are polymorphic. Polymorphic means a `WhatId` is equivalent to the ID of a related object. The label is `Related To ID`. This is a polymorphic relationship field.

Field	Details
	Relationship Name What **Relationship Type** Lookup **Refers To** Account, Accreditation, AssessmentIndicatorDefinition, AssessmentTask, AssessmentTaskContentDocument, AssessmentTaskDefinition, AssessmentTaskOrder, Asset, AssetRelationship, AssignedResource, Award, BoardCertification, BusinessLicense, BusinessMilestone, BusinessProfile, Campaign, CareBarrier, CareBarrierDeterminant, CareBarrierType, CareDeterminant, CareDeterminantType, CareDiagnosis, CareInterventionType, CareMetricTarget, CareObservation, CareObservationComponent, CarePgmProvHealthcareProvider, CarePreauth, CarePreauthItem, CareProgram, CareProgramCampaign, CareProgramEligibilityRule, CareProgramEnrollee, CareProgramEnrolleeProduct, CareProgramEnrollmentCard, CareProgramGoal, CareProgramProduct, CareProgramProvider, CareProgramTeamMember, CareProviderAdverseAction, CareProviderFacilitySpecialty, CareProviderSearchableField, CareRegisteredDevice, CareRequest, CareRequestDrug, CareRequestExtension, CareRequestItem, CareSpecialty, CareSpecialtyTaxonomy, CareTaxonomy, Case, CommSubscriptionConsent, ContactEncounter, ContactEncounterParticipant, ContactRequest, Contract, CoverageBenefit, CoverageBenefitItem, CreditMemo, DelegatedAccount, DocumentChecklistItem, EnrollmentEligibilityCriteria, HealthcareFacility, HealthcareFacilityNetwork, HealthcarePayerNetwork, HealthcarePractitionerFacility, HealthcareProvider, HealthcareProviderNpi, HealthcareProviderSpecialty, HealthcareProviderTaxonomy, IdentityDocument, Image, IndividualApplication, Invoice, ListEmail, Location, MemberPlan, Opportunity, Order, OtherComponentTask, PartyConsent, PersonLifeEvent, PlanBenefit, PlanBenefitItem, ProcessException, Product2, ProductItem, ProductRequest, ProductRequestLineItem, ProductTransfer, PurchaserPlan, ReceivedDocument, ResourceAbsence, ReturnOrder, ReturnOrderLineItem, ServiceAppointment, ServiceResource, Shift, Shipment, ShipmentItem, Solution, Visit, VisitedParty, VolunteerProject, WorkOrder, WorkOrderLineItem
`WhoCount`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** Available to organizations that have Shared Activities enabled. Represents the count of related EventRelations pertaining to the `WhoId`.
`WhoId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update

Field	Details
	Description The WhoId represents a human such as a lead or a contact. WhoIds are polymorphic. Polymorphic means a WhoId is equivalent to a contact's ID or a lead's ID. The label is `Name ID`. If Shared Activities is enabled, the value of this field is the ID of the related lead or primary contact. If you add, update, or remove the WhoId field, you might encounter problems with triggers, workflows, and data validation rules that are associated with the record. The label is `Name ID`. If the `JunctionIdList` field is used, all `WhoIds` are included in the relationship list. Beginning in API version 37.0, if the contact or lead ID in the `WhoId` field isn't in the `EventWhoIds` list, no error occurs and the ID is added to the `EventWhoIds` as the primary `WhoId`. If `WhoId` is set to null, an arbitrary ID from the existing `EventWhoIds` list is promoted to the primary position. This is a polymorphic relationship field. **Relationship Name** Who **Relationship Type** Lookup **Refers To** Contact, Lead

Usage

Use Event to manage calendar appointments.

Querying and Filtering Events

Queries on events are denied before they time out if they involve amounts of data that are deemed too large. In such cases, the exception code `OPERATION_TOO_LARGE` is returned. If you receive `OPERATION_TOO_LARGE`, refactor your query to return or scan a smaller amount of data.

When querying for events with a specific due date, you must filter on both the `ActivityDateTimeand` and `ActivityDate` fields. For example to find all events with a due date of February 14, 2003, you need two filters:

- One filter with the `ActivityDate` field equal to the Coordinated Universal Time (UTC) time zone on February 14, 2003.
- One filter with the `ActivityDate` field greater than or equal to midnight on February 14, 2003 in the user's local time zone AND less than or equal to midnight on February 15, 2003 in the user's local time zone.

Alternatively, in API version 13.0 and later, you can find events with a specific due date by filtering on `StartDateTime`. For example, to find all events with a due date of February 14, 2003, filter with the `StartDateTime` greater than or equal to midnight on February 14, 2003 in the user's local time zone AND less than or equal to midnight on February 15, 2003 in the user's local time zone.

The `EventId` field of an EventRelation object always points to the master record. An invitee on a group event can query the EventRelation object to view the master record.

Multiday Events

- Multiday events are available in API version 13.0 and later. Also, in earlier versions SOQL queries don't return multiday events.

- Multiday events are enabled through the user interface from Setup by entering `Activity Settings` in the `Quick Find` box, then selecting **Activity Settings**.
- If the multiday event feature is enabled, then API versions 13.0 and later support values greater than 1440 for the `DurationInMinutes` field. API versions 12.0 and earlier can't access event objects whose `DurationInMinutes` is greater than 1440.
- Multiday events can't exceed 14 days.

Event Series and Recurring Events

In Lightning Experience, events with multiple occurrences are called event series, and are indicated when the `IsRecurrence2` field is set to `true`. In Salesforce Classic, events with multiple occurrences are called recurring events, and are indicated when the `IsRecurrence` field is set to `true`. Both fields can't be set to true for the same event.

- Lightning Experience event series are available in API version 44.0 and later as read-only fields. Recurrence patterns, specified by the Recurrence2PatternText field, are creatable in API version 52.0 and later. Salesforce Classic recurring events are available in API version 7.0 and later. In earlier versions, SOQL queries don't return any Lightning Experience event series.
- After an event is created, you can't change the values of `IsRecurrence2` or `IsRecurrence` from `true` to `false` or vice versa.
- You can't set fields associated with `IsRecurrence2` for events where `IsRecurrence` is set to `true`, or vice versa.
- For Lightning Experience event series where `IsRecurrence2` is `true`, if you'd like to delete a single or all remaining events, use the REST API call. For Salesforce Classic recurring events where `IsRecurrence` is `true`, all past and future events in the series are removed when you delete the recurring event series through the API. However, when you delete the recurring event series through the user interface, only future occurrences are removed.
- For Lightning Experience event series in API version 58.0 and later, when you change a future event, events in the entire series also change. When you change a past event, `IsRecurrence2Exception` is set to `true` and only that past event changes.
- When creating a Salesforce Classic recurring event series, the duration of the event must be 24 hours or less. When the Salesforce Classic recurring event series is created, you can extend the length of individual occurrences beyond 24 hours if Multiday events are enabled; see **Multiday Events**.
- For Salesforce Classic recurring events, `RecurrenceStartDateTime`, `RecurrenceEndDateOnly`, `RecurrenceType`, and any properties associated with the given recurrence type (see the Recurrence Field Usage for Salesforce Classic Recurring Events table) must be populated.
- When updating a Salesforce Classic recurring event series, it's not possible to update the `EventRelation` for the event series object and the EventRelation for the series object occurrences at the same time.
- Lightning Experience event series have no series ID, so it's not possible to locate other occurrences in the series. In Salesforce Classic recurring events, you can use `RecurrenceActivityId` to locate other occurrences.
- For both Lightning Experience event series and Salesforce Classic recurring events, when a series repeats every day, month, or year, you can only schedule occurrences one time per day, month, or year. The week option lets you schedule occurrences multiple days per week.

Limits for Lightning Experience event series and limits for Salesforce Classic recurring events also apply.

Lightning Experience Event Series and Recurring Events

Use the `Recurrence2PatternText` field to specify the recurrence pattern for Lightning Experience event series. These recurrence patterns, called reference rules or RRULES, support a subset of the RFC 5545 standards. This table includes common RRULE examples.

Recurrence Pattern	RRULE Example
Every day for five days	`RRULE:FREQ=DAILY;INTERVAL=1;COUNT=5`

Recurrence Pattern	RRULE Example
Every Monday, Tuesday, Wednesday, Thursday, and Friday with no end date	`RRULE:FREQ=WEEKLY;INTERVAL=1;BYDAY=MO,TU,WE,TH,FR`
Every two weeks on Monday and Friday for 10 occurrences	`RRULE:FREQ=WEEKLY;INTERVAL=2;BYDAY=MO,FR;COUNT=10`
Monthly on the first day of the month until January 1, 2020	`RRULE:FREQ=MONTHLY;INTERVAL=1;BYMONTHDAY=1;UNTIL=20200101T100000Z`
Yearly on July 4th for three years (in this example, specify the date using StartDateTime)	`RRULE:FREQ=YEARLY;INTERVAL=1;BYMONTH=7;BYMONTHDAY=4;COUNT=3`
Daily until January 1, 2022 with no end date	`RRULE:FREQ=DAILY;UNTIL=20220101T000000Z`
Every third Friday of the month with no end date	`RRULE:FREQ=MONTHLY;BYSETPOS=3;BYDAY=FR`

The RRULE defined by `Recurrence2PatternText` supports a subset of the RFC 5545 standard for internet calendaring and scheduling. Supported RRULE parts include FREQ, BYMONTH, BYMONTHDAY, BYDAY, WKST, BYSETPOS, INTERVAL, UNTIL, and COUNT.

When the event record is saved, the RRULE might be modified to follow the required format:

- The RRULE parts are placed in the following order: FREQ, BYMONTH, BYMONTHDAY, BYDAY, WKST, BYSETPOS, INTERVAL, UNTIL, and COUNT.
- Any missing default values are inserted. For example, if the RRULE doesn't include INTERVAL, then `INTERVAL=1` is added.
- The RRULE is prefaced with `RRULE:` if that preface is missing.

RRULE Part	Supported RFC 5545 Implementation
FREQ	Required. Indicates the type of recurrence rule. Allowed values are: • `DAILY`—supported parts include FREQ, INTERVAL, UNTIL, and COUNT. • `WEEKLY`—supported parts include INTERVAL, UNTIL, COUNT, BYDAY, and WKST. BYDAY is required, but can't be preceded by a number. For example, to indicate weekly on Tuesday and Thursday until September 1, 2023, use `RRULE:FREQ=WEEKLY;UNTIL=20230901T000000Z;BYDAY=TU,TH` • `MONTHLY`—supported patterns include: – BYMONTHDAY For example, to indicate monthly on the third day of the month use: `RRULE:FREQ=MONTHLY;BYMONTHDAY=3` – BYDAY and BYSETPOS For example, to indicate the last weekday of the month, use `RRULE:FREQ=MONTHLY;BYDAY=MO,TU,WE,TH,FR;BYSETPOS=-1` – BYDAY, where the BYDAY values are specified with a numeric value For example, to indicate monthly on the first Friday for 10 occurrences, use `RRULE:FREQ=MONTHLY;COUNT=10;BYDAY=1FR` • `YEARLY`—supported patterns include: – BYMONTH, BYDAY, and BYSETPOS

RRULE Part	Supported RFC 5545 Implementation
	For example, to indicate every year on the second Friday of January, use `RRULE:FREQ=YEARLY;BYMONTH=1;BYDAY=FR;BYSETPOS=2` – BYMONTH and BYMONTHDAY For example, to indicate every year on October 31, use `RRULE:FREQ=YEARLY;BYMONTH=10;BYMONTHDAY=31` For example, to create a maintenance pattern such as twice in May, and September on 7th and 15th; and one time in June/July/August on the 1st, use two RRULEs: `RRULE: FREQ=MONTHLY; BYMONTH=5,9; BYMONTHDAY=7, 15` `RRULE:FREQ=MONTHLY; BYMONTH=6,7,8; BYMONTHDAY=1`
BYMONTH	The month. Valid values are `1` to `12`.
BYMONTHDAY	The day of the month. Valid values are `1` to `31`. If BYMONTHDAY is `31` and the month has fewer than 31 days, the event is created on the last day of the month.
BYDAY	A comma-separated list of days of the week. Valid values are `SU`, `MO`, `TU`, `WE`, `TH`, `FR`, `SA`. For RRULES with yearly or monthly frequency, BYDAY must be one of: • a single day • weekend days • weekdays • every day of the week Each BYDAY value can be preceded by an integer that indicates the nth occurrence of a specific day within the monthly or yearly RRULE. Allowed values are `-1`, `1`, `2`, `3`, and `4`. You can't use different numbers in the BYDAY values. For example, this RRULE isn't supported: `RRULE:FREQ=MONTHLY;INTERVAL=2;COUNT=10;BYDAY=1SU,-1SU` If BYDAY values are prefaced with a number, the RRULE can't include BYSETPOS.
WKST	Specifies the day on which the workweek starts. Valid values are `MO`, `TU`, `WE`, `TH`, `FR`, `SA`, and `SU`. Default value is based on the user's locale.
BYSETPOS	A comma-separated list of values that correspond to the nth occurrence within the set of recurrence instances specified by the rule. Valid values are `-1`, `1`, `2`, `3`, or `4`. Default value is `1`. For example, to indicate the last weekday of the month, use: `RRULE:FREQ=MONTHLY;BYDAY=MO,TU,WE,TH,FR;BYSETPOS=-1`
INTERVAL	The repetition interval. Valid values are: • an integer between `1` and `999` if FREQ=DAILY • an integer between `1` and `99` if FREQ=WEEKLY or FREQ=MONTHLY • `1` if FREQ=YEARLY Default value is `1`.

RRULE Part	Supported RFC 5545 Implementation
UNTIL	Specifies the datetime in UTC format when the recurrence rule stops. The supported format is yyyyMMddTHHmmssZ, for example: 20210419T083000Z. An RRULE can't contain both UNTIL and COUNT. A recurring event without either UNTIL or COUNT repeats indefinitely.
COUNT	The number of occurrences. Allowed values are 1 to 999. An RRULE can't contain both UNTIL and COUNT. A recurring event without either UNTIL or COUNT repeats indefinitely.
BYWEEKNO	Specifies a comma-separated list of values that specify weeks of the year. Valid values are: • `1 to 53` • `-53 to -1` For example, to indicate specific weeks in a year, use: `RRULE: BYWEEKNO=20, -20`. This rule part can't be used when the FREQ rule part is set to anything other than YEARLY. For example, 3 represents the third week of the year. Note: Assuming a Monday week start, week 53 can only occur when Thursday is January 1 or if it's a leap year and Wednesday is January 1.
BYYEARDAY	Specifies a comma-separated list of values that specify days of the year. Valid values are: • `1 to 366` • `-366 to -1` For example, to indicate specific days in a year, use: `RRULE:BYYEARDAY=1,100,200`; or, `RRULE:BYYEARDAY=1,-2`.

Salesforce Classic Event Series and Recurring Events

This table describes the usage of recurrence fields for Salesforce Classic recurring events. Each recurrence type must have all of its properties set. All unused properties must be set to null.

RecurrenceType Value	Properties	Example Pattern
RecursDaily	RecurrenceInterval	Every second day
RecursEveryWeekday	RecurrenceDayOfWeekMask	Every weekday - can't be Saturday or Sunday
RecursMonthly	RecurrenceDayOfMonth RecurrenceInterval	Every second month, on the third day of the month
RecursMonthlyNth	RecurrenceInterval RecurrenceInstance RecurrenceDayOfWeekMask	Every second month, on the last Friday of the month
RecursWeekly	RecurrenceInterval RecurrenceDayOfWeekMask	Every three weeks on Wednesday and Friday
RecursYearly	RecurrenceDayOfMonth RecurrenceMonthOfYear	Every March on the 26th day of the month

RecurrenceType Value	Properties	Example Pattern
RecursYearlyNth	RecurrenceDayOfWeekMask RecurrenceInstanceRecurrenceMonthOfYear	The first Saturday in every October

Attendees, Invitees, and Resources

The field `GroupEventType` indicates that event participants are included on an event. You can add a resource to an event only when the resource is available. The only attendance status that can be assigned to resources is Accepted. Events can't be saved when resources you've added aren't available.

JunctionIdList

To create an event using `JunctionIdList`, IDs are pulled from the related contacts and both the event and the `EventRelation` records are created in one API call. If the `EventRelation` fails, the event is rolled back because it's all done in a single API call.

```
public void createEventNew(Contact[] contacts) {
 String[] contactIds = new String[contacts.size()];
 for (int i = 0; i < contacts.size(); i++) {
  contactIds[i] = contacts[i].getID();
 }
 Event event = new Event();
 event.setSubject("New Event");
 event.setEventWhoIds(contactIds);
 SaveResult[] results = null;
 try {
  results = connection.create(new Event[] {
   task
  });
 } catch (ConnectionException ce) {
  ce.printStackTrace();
 }
}
```

Syncing Events with Lightning Sync

Attendee statuses (Accepted or Maybe, Declined, or No Response) sync from Microsoft® Exchange or Google to Salesforce, but not from Salesforce to Exchange or Google. Be wary of creating API flows that update attendee status in Salesforce for users set up to sync both ways. Eventually the original Exchange or Google status overrides the update made in Salesforce.

Shared Field-Level Security for Event and Task Objects

Metadata deployments for the Event object must include the field-level security for the Task object. Shared field-level security prevents each object from changing the field-level security of the associated object.

Metadata deployments that include field-level security for only one of either the Event or Task objects can cause field-level security changes to the other object that aren't reflected in the metadata.

- If field-level security is enabled for one object, then field-level security is enabled for both objects.
- If field-level security is disabled for one object, then it's disabled for both objects.

Note: A missing entry in the metadata is treated as field-level security being disabled.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

EventChangeEvent (API version 44.0)
Change events are available for the object.

EventFeed (API version 20.0)
Feed tracking is available for the object.

Task

Represents a business activity such as making a phone call or other to-do items. In the user interface, Task and Event records are collectively referred to as activities.

Note: Task fields related to calls are exclusive to Salesforce CRM Call Center. Also, `query()`, `delete()`, and `update()` aren't allowed with tasks related to more than one contact in API versions 23.0 and earlier.

Supported Calls

`create()`, `delete()`, `describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `undelete()`, `update()`, `upsert()`

Fields

Field	Field Type
`AccountId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** Represents the ID of the related Account. The `AccountId` is determined as follows. If the value of `WhatId` is any of the following objects, then Salesforce uses that object's `AccountId`. • Account • Opportunity • Contract • Custom object that is a child of Account If the value of the `WhatId`field is any other object, and the value of the `WhoId` field is a Contact object, then Salesforce uses that contact's `AccountId`. (If your organization uses Shared Activities, then Salesforce uses the `AccountId` of the primary contact.) Otherwise, Salesforce sets the value of the `AccountId` field to `null`. For information on IDs, see ID Field Type.

Field	Field Type
	This is a relationship field. **Relationship Name** Account **Relationship Type** Lookup **Refers To** Account
`ActivityDate`	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Represents the due date of the task. This field has a timestamp that is always set to midnight in the Coordinated Universal Time (UTC) time zone. The timestamp is not relevant; do not attempt to alter it to accommodate time zone differences. Label is **Due Date**. Note: This field can't be set or updated for a recurring task (`IsRecurrence` is `true`).
`CallDisposition`	**Type** string **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Represents the result of a given call, for example, "we'll call back," or "call unsuccessful." Limit is 255 characters. Not subject to field-level security, available for any user in an organization with Salesforce CRM Call Center.
`CallDurationInSeconds`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** Duration of the call in seconds. Not subject to field-level security, available for any user in an organization with Salesforce CRM Call Center.
`CallObject`	**Type** string

Field	Field Type
	Properties Create, Filter, Group, Nillable, Sort, Update **Description** Name of a call center. Limit is 255 characters. Not subject to field-level security, available for any user in an organization with Salesforce CRM Call Center.
`CallType`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The type of call being answered: Inbound, Internal, or Outbound.
`CompletedDateTime`	**Type** dateTime **Properties** Filter, Nillable, Sort **Description** The date and time the task was saved with a Closed status. • For insert, if the task is saved with a Closed status the field is set. If the task is saved with an Open status the field is set to NULL. • For update, if the task is saved with a new Closed status, the field is reset. If the task is saved with a new non-closed status, the field is reset to NULL. If the task is saved with the same closed status (that is, unchanged) there is no change to the field. Note: The status is a dynamic enum. If the Closed mapping is changed it won't cause an update of existing tasks. Only new insert/update operations are affected.
`ConnectionReceivedId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** ID of the PartnerNetworkConnection that shared this record with your organization. This field is available if you enabled Salesforce to Salesforce.
`ConnectionSentId`	**Type** reference

Field	Field Type
	Properties Filter, Group, Nillable, Sort **Description** ID of the PartnerNetworkConnection that you shared this record with. This field is available if you enabled Salesforce to Salesforce. This field is supported using API versions earlier than 15.0. In all other API versions, this field's value is null. You can use the new PartnerNetworkRecordConnection object to forward records to connections.
`Description`	**Type** textarea **Properties** Create, Nillable, Update **Description** Contains a text description of the task.
`IsArchived`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Indicates whether the event has been archived. The default value of this field is `false`.
`IsClosed`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Indicates whether the task has been completed (`true`) or not (`false`). The default value of this field is `false`. Is only set indirectly via the `Status` picklist. Label is **Closed**.
`IsHighPriority`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Indicates a high-priority task. This field is derived from the `Priority` field. The default value of this field is `false`.
`IsRecurrence`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort

Field	Field Type
	Description Indicates whether the task is scheduled to repeat itself (`true`) or only occurs once (`false`). The default value of this field is `false`. This field is read-only on update, but not on create. If this field value is `true`, then `RecurrenceStartDateOnly`, `RecurrenceEndDateOnly`, `RecurrenceType`, and any recurrence fields associated with the given recurrence type must be populated. See Recurring Tasks.
`IsReminderSet`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether a popup reminder has been set for the task (`true`) or not (`false`). The default value of this field is `false`.
`IsVisibleInSelfService`	**Type** boolean **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Indicates whether a task associated with an object can be viewed in the Customer Portal (`true`) or not (`false`). If your organization has digital experiences enabled, tasks marked `IsVisibleInSelfService` are visible to any external user in the Experience Cloud site, as long as the user has access to the record the task was created on.
`OwnerId`	**Type** reference **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** ID of the User or Group who owns the record. Label is **Assigned To ID**. This field accepts Groups of type Queue only. In the user interface, Group IDs correspond with the queue's list view names. To create or update tasks assigned to Group, use v48.0 or later. This is a polymorphic relationship field. **Relationship Name** Owner **Relationship Type** Lookup **Refers To** Group, User

Field	Field Type
`Priority`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. Indicates the importance or urgency of a task, such as high or low. The default value of this field is `Normal`.
`RecurrenceActivityId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** Read-only. Not required on create. ID of the main record of the recurring task. Subsequent occurrences have the same value in this field.
`RecurrenceDayOfMonth`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The day of the month in which the task repeats.
`RecurrenceDayOfWeekMask`	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The day or days of the week on which the task repeats. This field contains a bitmask. The values are as follows: • Sunday = 1 • Monday = 2 • Tuesday = 4 • Wednesday = 8 • Thursday = 16 • Friday = 32 • Saturday = 64 Multiple days are represented as the sum of their numerical values. For example, Tuesday and Thursday = 4 + 16 = 20.

Field	Field Type
RecurrenceEndDateOnly	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The last date on which the task repeats. This field has a timestamp that is always set to midnight in the Coordinated Universal Time (UTC) time zone. The timestamp is not relevant; do not attempt to alter it to accommodate time zone differences.
RecurrenceInstance	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The frequency of the recurring task. Possible values are: • First—1st • Fourth—4th • Last—last • Second—2nd • Third—3rd
RecurrenceInterval	**Type** int **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The interval between recurring tasks.
RecurrenceMonthOfYear	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The month of the year in which the task repeats.
RecurrenceRegeneratedType	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update

Field	Field Type
	Description Represents what triggers a repeating task to repeat. Add this field to a page layout together with the `RecurrenceInterval` field, which determines the number of days between the triggering date (due date or close date) and the due date of the next repeating task in the series. Label is **Repeat This Task**. This field has the following picklist values: • **None**: The task doesn't repeat. • **After due date**: The next repeating task will be due the specified number of days after the current task's due date. • **After the task is closed**: The next repeating task will be due the specified number of days after the current task is closed. • **(Task closed)**: This task, now closed, was opened as part of a repeating series. Note: When tasks in a series are set to repeat after their due date, Salesforce doesn't create recurrences that would have been due in the past. Instead, Salesforce keeps adding the interval until a repeated task has a due date in the future. For example, suppose that someone sets a task to repeat three days after it's due. But, that person doesn't complete the task (mark it Closed) until five days after it's due. Instead of creating a task that's already overdue, Salesforce gives the new task a due date of tomorrow. This due date is equivalent to 6 days after the due date; two intervals of three days each. If that person completes the repeating task (marks it Closed) before the due date, the next task is still due three days after the due date.
`RecurrenceStartDateOnly`	**Type** date **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The date when the recurring task begins. Must be a date and time before `RecurrenceEndDateOnly`.
`RecurrenceTimeZoneSidKey`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** The time zone associated with the recurring task. For example, "UTC-8:00" for Pacific Standard Time.
`RecurrenceType`	**Type** picklist

Field	Field Type
	Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update **Description** Indicates how often the task repeats. For example, daily, weekly, or every nth month (where "nth" is defined in `RecurrenceInstance`).
`ReminderDateTime`	**Type** dateTime **Properties** Create, Filter, Nillable, Sort, Update **Description** Represents the time when the reminder is scheduled to fire, if `IsReminderSet` is set to `true`. If `IsReminderSet` is set to `false`, then the user may have deselected the reminder checkbox in the Salesforce user interface, or the reminder has already fired at the time indicated by the value.
`Status`	**Type** picklist **Properties** Create, Defaulted on create, Filter, Group, Sort, Update **Description** Required. Indicates the status of the task. The default value of this field is `Not Started`. Each predefined `Status` field implies a value for the `IsClosed` flag. To obtain picklist values, query the TaskStatus object. Possible values are: • Completed • Deferred • In Progress • Not Started • Waiting on someone else Note: This field can't be updated for recurring tasks (`IsRecurrence` is `true`).
`Subject`	**Type** combobox **Properties** Create, Filter, Nillable, Sort, Update **Description** The subject line of the task, such as "Call" or "Send Quote." Limit: 255 characters.
`TaskSubtype`	**Type** picklist

Field	Field Type
	Properties Create, Filter, Group, Nillable, Restricted picklist, Sort **Description** Provides standard subtypes to facilitate creating and searching for specific task subtypes. This field isn't updateable. `TaskSubtype` values: • `Task` • `Email` • `LinkedIn` —Available in API version 56.0 and later. • `List Email` • `Cadence` • `Call` Note: The `Cadence` subtype is an internal value used by Sales Engagement, and can't be set manually.
`TaskWhoIds`	**Type** JunctionIdList **Properties** Create, Update **Description** A string array of contact or lead IDs related to this task. This `JunctionIdList` field is linked to the `TaskWhoRelations` child relationship. `TaskWhoIds` is only available when the shared activities setting is enabled. The first contact or lead ID in the list becomes the primary `WhoId` if you don't specify a primary `WhoId`. If you set the `EventWhoIds` field to null, all entries in the list are deleted and the value of `WhoId` is added as the first entry. Warning: Adding a `JunctionIdList` field name to the `fieldsToNull` property deletes all related junction records. This action can't be undone.
`Type`	**Type** picklist **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The type of task, such as Call or Meeting.
`WhatCount`	**Type** int **Properties** Filter, Group, Nillable, Sort

Field	Field Type
	Description Available to organizations that have Shared Activities enabled. Count of related TaskRelations pertaining to `WhatId`. Count of the `WhatId` must be *1* or less.
`WhatId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The `WhatId` represents nonhuman objects such as accounts, opportunities, campaigns, cases, or custom objects. `WhatId`s are polymorphic. Polymorphic means a `WhatId` is equivalent to the ID of a related object. The label is `Related To ID`. This is a polymorphic relationship field. **Relationship Name** What **Relationship Type** Lookup **Refers To** Account, Accreditation, AssessmentIndicatorDefinition, AssessmentTask, AssessmentTaskContentDocument, AssessmentTaskDefinition, AssessmentTaskOrder, Asset, AssetRelationship, AssignedResource, Award, BoardCertification, BusinessLicense, BusinessMilestone, BusinessProfile, Campaign, CareBarrier, CareBarrierDeterminant, CareBarrierType, CareDeterminant, CareDeterminantType, CareDiagnosis, CareInterventionType, CareMetricTarget, CareObservation, CareObservationComponent, CarePgmProvHealthcareProvider, CarePreauth, CarePreauthItem, CareProgram, CareProgramCampaign, CareProgramEligibilityRule, CareProgramEnrollee, CareProgramEnrolleeProduct, CareProgramEnrollmentCard, CareProgramGoal, CareProgramProduct, CareProgramProvider, CareProgramTeamMember, CareProviderAdverseAction, CareProviderFacilitySpecialty, CareProviderSearchableField, CareRegisteredDevice, CareRequest, CareRequestDrug, CareRequestExtension, CareRequestItem, CareSpecialty, CareSpecialtyTaxonomy, CareTaxonomy, Case, CommSubscriptionConsent, ContactEncounter, ContactEncounterParticipant, ContactRequest, Contract, CoverageBenefit, CoverageBenefitItem, CreditMemo, DelegatedAccount, DocumentChecklistItem, EnrollmentEligibilityCriteria, HealthcareFacility, HealthcareFacilityNetwork, HealthcarePayerNetwork, HealthcarePractitionerFacility, HealthcareProvider, HealthcareProviderNpi, HealthcareProviderSpecialty, HealthcareProviderTaxonomy, IdentityDocument, Image, IndividualApplication, Invoice, ListEmail, Location, MemberPlan, Opportunity, Order, OtherComponentTask, PartyConsent, PersonLifeEvent, PlanBenefit, PlanBenefitItem, ProcessException, Product2, ProductItem, ProductRequest, ProductRequestLineItem, ProductTransfer, PurchaserPlan, ReceivedDocument, ResourceAbsence, ReturnOrder, ReturnOrderLineItem, ServiceAppointment, ServiceResource, Shift, Shipment, ShipmentItem, Solution, Visit, VisitedParty, VolunteerProject, WorkOrder, WorkOrderLineItem

Field	Field Type
`WhoCount`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** Available to organizations that have Shared Activities enabled. Count of related TaskRelations pertaining to `WhoId`.
`WhoId`	**Type** reference **Properties** Create, Filter, Group, Nillable, Sort, Update **Description** The WhoId represents a human such as a lead or a contact. WhoIds are polymorphic. Polymorphic means a WhoId is equivalent to a contact's ID or a lead's ID. The label is `Name ID`. If Shared Activities is enabled, the value of this field is the ID of the related lead or primary contact. If you add, update, or remove the WhoId field, you might encounter problems with triggers, workflows, and data validation rules that are associated with the record. The label is `Name ID`. Beginning in API version 37.0, if the contact or lead ID in the `WhoId` field is not in the `TaskWhoIds` list, no error occurs and the ID is added to the `TaskWhoIds` as the primary `WhoId`. If `WhoId` is set to null, an arbitrary ID from the existing `TaskWhoIds` list is promoted to the primary position. This is a polymorphic relationship field. **Relationship Name** Who **Relationship Type** Lookup **Refers To** Contact, Lead

Usage

Recurring Tasks

- Recurring tasks are available in API version 16.0 and later.
- After a task is created, it can't be changed from recurring to nonrecurring or vice versa.
- When a user creates a series of recurring tasks, Salesforce creates a main record and subsequent occurrences. For the main record, `IsRecurrence` is set to `true` and other fields that define the recurrence pattern are populated. The ID of the main record of the recurring task is saved in the subsequent occurrences, in the `RecurrenceActivityId` field.

- When you delete a recurring task series through the API, all open and closed task occurrences in the series are removed. However, when you delete a recurring task series through the user interface, only open tasks occurrences (`IsClosed` is `false`) in the series are removed.
- If `IsRecurrence` is `true`, then `RecurrenceStartDateOnly`, `RecurrenceEndDateOnly`, `RecurrenceType`, and any properties associated with the given recurrence type (see the following table) must be populated.
- When you change the `RecurrenceStartDateOnly` field or the recurrence pattern, all open tasks occurrences in the series are deleted and new open task occurrences are created based on the new recurrence pattern. The following fields determine the recurrence pattern: `RecurrenceType`, `RecurrenceTimeZoneSidKey`, `RecurrenceInterval`, `RecurrenceDayOfWeekMask`, `RecurrenceDayOfMonth`, `RecurrenceInstance`, and `RecurrenceMonthOfYear`.
- When you change the value of `RecurrenceEndDateOnly` to an earlier date (for example, from January 20 to January 10), all open task occurrences in the series with the `ActivityDate` value greater than the new end date value are deleted. Other open and closed task occurrences in the series are not affected.
- When you change the value of `RecurrenceEndDateOnly` to a later date (for example, from January 10 to January 20), new task occurrences are created up to the new end date. Existing open and closed tasks in the series are not affected.

This table describes the usage of recurrence fields for Salesforce Classic recurring events. Each recurrence type must have all of its properties set. All unused properties must be set to null.

RecurrenceType Value	Properties	Example Pattern
RecursDaily	RecurrenceInterval	Every second day
RecursEveryWeekday	RecurrenceDayOfWeekMask	Every weekday - can't be Saturday or Sunday
RecursMonthly	RecurrenceDayOfMonth RecurrenceInterval	Every second month, on the third day of the month
RecursMonthlyNth	RecurrenceInterval RecurrenceInstance RecurrenceDayOfWeekMask	Every second month, on the last Friday of the month
RecursWeekly	RecurrenceInterval RecurrenceDayOfWeekMask	Every three weeks on Wednesday and Friday
RecursYearly	RecurrenceDayOfMonth RecurrenceMonthOfYear	Every March on the 26th day of the month
RecursYearlyNth	RecurrenceDayOfWeekMask RecurrenceInstanceRecurrenceMonthOfYear	The first Saturday in every October

JunctionIdList

The `JunctionIdList` field is now implemented in the Event and Task objects. With a single API call, it's easy to create many-to-many relationships between the Event or Task object with contacts, leads, or users.

To create a Task with related Contacts without `JunctionIdList`, you first have to create the task, then use the returned task ID to create the `TaskRelation` records. If the `TaskRelation` save call fails, error handling is your responsibility because the task has already been committed to the database.

```
public void createTasksOld(Contact[] contacts) {
 Task task = new Task();
 task.setSubject("New Task");
 SaveResult[] results = null;
```

```
 try {
  results = connection.create(new Task[] {
   task
  });
  if (results[0].isSuccess()) {
   TaskRelation[] relations = new TaskRelation[contacts.size()];
   for (int i = 0; i < contacts.length; i++) {
    relations[i] = new TaskRelation();
    relations[i].setTaskId(results[0].getID());
    relations[i].setRelationId(contacts[i].getID());
   }
   results = connection.create(relations);
  }
 } catch (ConnectionException ce) {
  ce.printStackTrace();
 }
}
```

To create a task using `JuncionIdList`, IDs are pulled from the related contacts and both the task and the `TaskRelation` records are created in one API call. If the `TaskRelation` fails, the task is rolled back because it's all done in a single API call.

```
public void createTaskNew(Contact[] contacts) {
 String[] contactIds = new String[contacts.size()];
 for (int i = 0; i < contacts.size(); i++) {
  contactIds[i] = contacts[i].getID();
 }
 Task task = new Task();
 task.setSubject("New Task");
 task.setTaskWhoIds(contactIds);
 SaveResult[] results = null;
 try {
  results = connection.create(new Task[] {
   task
  });
 } catch (ConnectionException ce) {
  ce.printStackTrace();
 }
}
```

Shared Field-Level Security for Event and Task Objects

Metadata deployments for the Task object should always include the field-level security for the Event object. Shared field-level security prevents each object from changing the field-level security of the associated object.

Metadata deployments that include field-level security for only one of either the Event or Task objects can cause field-level security changes to the other object that aren't reflected in the metadata.

- If field-level security is enabled for one object, then field-level security is enabled for both objects.
- If field-level security is disabled for one object, then it's disabled for both objects.

Note: A missing entry in the metadata is treated as field-level security being disabled.

Associated Objects

This object has the following associated objects. If the API version isn't specified, they're available in the same API versions as this object. Otherwise, they're available in the specified API version and later.

TaskChangeEvent (API version 44.0)

Change events are available for the object.

TaskFeed (API version 20.0)

Feed tracking is available for the object.