



---

# Loyalty Management Developer Guide

Version 63.0, Spring '25





# CONTENTS

<b>Chapter 1: Loyalty Management Standard Objects</b> .....	<b>1</b>
Loyalty Management Data Model .....	3
Benefit .....	4
BenefitParameterValue .....	7
BenefitType .....	8
GameDefinition .....	11
GameParticipant .....	14
GameParticipantReward .....	18
GameReward .....	21
JournalReason .....	24
JournalSubType .....	25
JournalType .....	27
LoyaltyAggrPointExprLedger .....	29
LoyaltyLedger .....	31
LoyaltyLedgerTraceability .....	35
LoyaltyMemberCurrency .....	38
LoyaltyMembershipLifecycle .....	43
LoyaltyMemberTier .....	46
LoyaltyPartnerProduct .....	49
LoyaltyPgmCurrencySubtype .....	52
LoyaltyPgmEngmtAttribute .....	54
LoyaltyPgmEngmtAttrProm .....	57
LoyaltyPgmGroupMbrRInsp .....	59
LoyaltyPgmMbrAttributeVal .....	62
LoyaltyPgmMbrPromEligView .....	65
LoyaltyPgmPartnerCurrency .....	73
LoyaltyPgmPartnerPromotion .....	75
LoyaltyPgmPtrnLdgrSummary .....	77
LoyaltyPgmPtrnPrepaidPack .....	79
LoyaltyProgram .....	82
LoyaltyProgramBadge .....	84
LoyaltyProgramCurrency .....	87
LoyaltyProgramCurrencyTier .....	91
LoyaltyProgramMbrPromotion .....	93
LoyaltyProgramMember .....	96
LoyaltyProgramMemberBadge .....	103
LoyaltyProgramMemberCase .....	105
LoyaltyProgramMemberMerge .....	107
LoyaltyProgramPartner .....	113

## Contents

LoyaltyProgramPartnerLedger	119
LoyaltyProgramProcess	122
LoyaltyProgramWidget	126
LoyaltyTier	128
LoyaltyTierBenefit	131
LoyaltyTierGroup	132
MemberBenefit	136
NetworkReferencedObject	140
Promotion	142
PromotionActionableList	153
PromotionChannel	154
PromotionLimit	157
PromotionLimitUsage	160
PromotionLoyaltyPtnrProdt	162
PromotionMarketSegment	165
PromotionPartyUsage	167
PromotionPartyUsageDetail	171
PromotionProduct	174
PromotionProductCategory	176
PromotionRuleTemplate	178
PromotionStage	180
PromotionStageEmailTemplate	182
TransactionJournal	184
Voucher	198
VoucherDefinition	205
<b>Chapter 2: Loyalty Management Fields on Standard Objects</b>	<b>212</b>
Loyalty Management Fields on EngagementChannelType	213
Loyalty Management Fields on InternalOrganizationUnit	213
<b>Chapter 3: Coupon Objects in Loyalty Management</b>	<b>215</b>
<b>Chapter 4: Loyalty Management Associated Objects</b>	<b>216</b>
StandardObjectNameChangeEvent	217
<b>Chapter 5: Apex Reference</b>	<b>220</b>
ConnectApi Namespace	221
LoyaltyConnect Methods	221
LoyaltyManagementConnect Methods	222
LoyaltyVoucherConnect Methods	226
Input Classes	230
Output Classes	235
ConnectApi Enums	242
LoyaltyManagement Namespace	243
CdpBasedLoyaltyPromotionInput Class	245

## Contents

CdpBasedLoyaltyPromotionInputBuilder Class	246
CdpBasedLoyaltyPromotionOutput Class	247
ChangeTierInput Class	248
ChangeTierInputBuilder Class	249
ChangeTierOutput Class	252
CreditPointsOutput Class	254
DebitPointsOutput Class	256
GetMemberActiveSegmentsInput Class	258
GetMemberActiveSegmentsInputBuilder Class	258
GetMemberActiveSegmentsOutput Class	260
IssueVoucherInput Class	261
IssueVoucherInputBuilder Class	261
IssueVoucherOutput Class	267
LoyaltyActionError Class	269
LoyaltyActionResult Class	270
LoyaltyPromotionInput Class	271
LoyaltyPromotionInputBuilder Class	272
LoyaltyPromotionOutput Class	273
LoyaltyResources Class	274
MemberPointBalanceInput Class	279
MemberPointBalanceInputBuilder Class	280
MemberPointBalanceOutput Class	281
MemberTierInput Class	284
MemberTierInputBuilder Class	285
MemberTierOutput Class	286
PointsInput Class	289
PointsInputBuilder Class	290
TransferMemberPointsToGroupsInput Class	294
TransferMemberPointsToGroupsInputBuilder Class	295
UpdateCumulativeUsageCompletedInput Class	296
UpdateCumulativeUsageCompletedOutput Class	297
<b>Chapter 6: Loyalty Management Business APIs</b>	<b>300</b>
REST Reference	301
Resources	301
Request Bodies	353
Response Bodies	374
Loyalty Management Connect API Interchangeable Request Fields	445
<b>Chapter 7: Loyalty Management Standard Invocable Actions</b>	<b>447</b>
Adjust Points	449
Assign Member Tier Benefits	451
Accrual Cancellation	452
Credit Points	454

## Contents

Change Tier	456
Change All Tiers or None	459
Debit Points	461
Generate Referral Code	464
Get Member's Active Segments	465
Get Points Balance	467
Get Promotions Based on Member's Market Segment	469
Get Promotions Based on Transaction Journal	471
Get Tier	472
Issue Voucher	474
Merge Memberships Action	478
Process Member Benefit	480
Redemption Cancellation	481
Run Program Process for Transaction Journal	483
Transfer Member Points to Groups	484
Unmerge Memberships Action	485
Update Member Progress for Cumulative Promotion	487
<b>Chapter 8: Loyalty Management Custom Invocable Actions</b>	<b>490</b>
Run Program Process	491
<b>Chapter 9: Loyalty Management Metadata API Types</b>	<b>493</b>
BenefitAction	494
LoyaltyProgramSetup	496
Settings	511
IndustriesGamificationSettings	511
IndustriesLoyaltySettings	512
IndustriesUnifiedPromotionsSettings	515
<b>Chapter 10: Loyalty Management Tooling API Objects</b>	<b>518</b>
BenefitAction	519
<b>Chapter 11: Loyalty Management Mobile SDK</b>	<b>524</b>
What to Expect from Loyalty Management Mobile SDK	525
Supported Versions of Tools and Components	525
Supported DateTime Format	526
Getting Started with Loyalty Management Mobile SDK	526
Prepare to Install Loyalty Management Mobile SDK	527
Create a Connected App to Integrate with Loyalty Management	527
Prepare Experience Cloud Users for Loyalty Management Mobile SDK	528
Loyalty Management Mobile SDK for iOS Development	530
Clone and Import Loyalty Management Mobile SDK for iOS	531
Loyalty Management Mobile SDK for iOS Classes	532
Loyalty Management Mobile SDK API References for iOS	545
Loyalty Management Mobile SDK for Android Development	574

## Contents

Clone and Import Loyalty Management Mobile SDK for Android . . . . .	575
Loyalty Management Mobile SDK for Android Classes . . . . .	576
Loyalty Management Mobile SDK API References for Android . . . . .	587
Loyalty Management Sample App for Mobile SDK . . . . .	587
Use the Loyalty Management Mobile SDK Sample App . . . . .	588
Walkthrough of the Sample App for Loyalty Management Mobile SDK . . . . .	589
Customize the Android SDK . . . . .	589
Customize the iOS SDK . . . . .	590
Receipt Scanning with Loyalty Management Mobile SDK . . . . .	592
How Receipt Scanning Works . . . . .	592
Loyalty Management Receipt Scanner Kit Unlocked Package . . . . .	596
Set Up Amazon Web Services . . . . .	599
Set Up Salesforce Org With Loyalty Management Receipt Scanner Kit Unlocked Package . . . . .	604
Receipt Scan and Upload on iOS and Android . . . . .	610
Loyalty Management Mobile SDK and Sample App Support . . . . .	613
Upgrade Loyalty Management Mobile SDK for iOS . . . . .	613
<b>Chapter 12: Gamification Mobile SDK . . . . .</b>	<b>614</b>
Supported Versions of Tools and Components . . . . .	615
Gamification Mobile SDK for iOS . . . . .	615
Clone and Import Gamification Mobile SDK for iOS . . . . .	616
Integrate Gamification Mobile iOS SDK into Loyalty Management Mobile iOS SDK . . . . .	616
Gamification Mobile SDK iOS Classes . . . . .	617
Gamification Mobile SDK API Reference for iOS . . . . .	620
Game Zone UI Customization for iOS . . . . .	628
Gamification Mobile SDK for Android . . . . .	633
Clone and Import Gamification Mobile SDK for Android . . . . .	634
Integrate Gamification Mobile Android SDK into Loyalty Management Mobile Android SDK . . . . .	635
Gamification Mobile SDK Android Classes . . . . .	635
Gamification Mobile Android SDK API Reference . . . . .	638
Game Zone UI Customization for Android . . . . .	638
<b>Chapter 13: Data Processing Engine, Batch Management, and Monitor Workflow Services . . . . .</b>	<b>640</b>
Data Model . . . . .	641
Common Tooling API Object . . . . .	642
BatchJobDefinition . . . . .	642
Common Platform Event . . . . .	648
BatchJobStatusChangedEvent . . . . .	648
Common Business APIs . . . . .	650
REST Reference . . . . .	650
Data Processing Engine . . . . .	652

## Contents

Data Processing Engine Tooling API Objects .....	653
Data Processing Engine Standard Object .....	658
Data Processing Engine Metadata API .....	663
Data Processing Engine Invocable Actions .....	688
Batch Management .....	691
Batch Management Tooling API Objects .....	691
Batch Management Standard Objects .....	697
Batch Management Metadata API .....	699
Batch Management Invocable Actions .....	705
Monitor Workflow Services .....	708
BatchJob .....	708
BatchJobPart .....	712
BatchJobPartFailedRecord .....	716



# CHAPTER 1 Loyalty Management Standard Objects

## In this chapter ...

- [Loyalty Management Data Model](#)
- [Benefit](#)
- [BenefitParameterValue](#)
- [BenefitType](#)
- [GameDefinition](#)
- [GameParticipant](#)
- [GameParticipantReward](#)
- [GameReward](#)
- [JournalReason](#)
- [JournalSubType](#)
- [JournalType](#)
- [LoyaltyAggrPointExpLdger](#)
- [LoyaltyLedger](#)
- [LoyaltyLedgerTraceability](#)
- [LoyaltyMemberCurrency](#)
- [LoyaltyMembershipLifecycle](#)
- [LoyaltyMemberTier](#)
- [LoyaltyPartnerProduct](#)
- [LoyaltyPgmCurrencySubtype](#)
- [LoyaltyPgmEngmtAttribute](#)
- [LoyaltyPgmEngmtAttrProm](#)
- [LoyaltyPgmGroupMbrRhsp](#)
- [LoyaltyPgmMbrAttributeVal](#)
- [LoyaltyPgmMbrPromBgView](#)
- [LoyaltyPgmPartnerCurrency](#)
- [LoyaltyPgmPartnerPromotion](#)
- [LoyaltyPgmPntLdgSummary](#)
- [LoyaltyPgmPntPrepaidPack](#)
- [LoyaltyProgram](#)
- [LoyaltyProgramBadge](#)
- [LoyaltyProgramCurrency](#)
- [LoyaltyProgramCurrencyTier](#)

Loyalty Management data model provides objects and fields to create and manage loyalty programs for your organization. Use the objects to manage partners and members, track members' activities, reward members, and manage promotions.

In addition to the standard objects included with the Loyalty Management license, the standard objects included with the Salesforce platform licenses enabled in your Org are available out-of-the-box.



**Warning:** When you customize standard fields, we recommend that you don't change the API names of the default picklist values.

## SEE ALSO:

[Object Reference for the Salesforce Platform: Overview of Salesforce Platform Objects and Fields](#)  
[SOAP API Developer Guide: About SOAP API](#)

## Loyalty Management Standard Objects

- LoyaltyProgramViaPromotion
- LoyaltyProgramMember
- LoyaltyProgramMemberBadge
- LoyaltyProgramMemberCase
- LoyaltyProgramMemberMerge
- LoyaltyProgramPartner
- LoyaltyProgramPartnerLedger
- LoyaltyProgramProcess
- LoyaltyProgramWidget
- LoyaltyTier
- LoyaltyTierBenefit
- LoyaltyTierGroup
- MemberBenefit
- NetworkReferencedObject
- Promotion
- PromotionActionableList
- PromotionChannel
- PromotionLimit
- PromotionLimitUsage
- PromotionLoyaltyPtnrProdt
- PromotionMarketSegment
- PromotionPartyUsage
- PromotionPartyUsageDetail
- PromotionProduct
- PromotionProductCategory
- PromotionRuleTemplate
- PromotionStage
- PromotionStageEmailTemplate
- TransactionJournal
- Voucher
- VoucherDefinition



View the Loyalty Management's [Loyalty Journal data model](#) on the Salesforce Architect page.

## Benefit

---

Represents information about benefits associated with the loyalty program. Benefits are mapped to the members based on the tier or other program rules. This object is available in API version 51.0 and later.

## Supported Calls

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

## Fields

Field	Details
BenefitActionId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> ID of the benefit action. This is a relationship field.</p> <p><b>Relationship Name</b> BenefitAction</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> BenefitAction</p>
BenefitTypeId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> ID of the benefit type. This is a relationship field.</p> <p><b>Relationship Name</b> BenefitType</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> BenefitType</p>

Field	Details
Description	<p><b>Type</b> textarea</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> Detailed description of loyalty benefit.</p>
ImageUrl	<p><b>Type</b> url</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The URL for the image. This field is available in API version 57.0 and later.</p>
IsActive	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. Indicates whether the loyalty benefit is active.</p>
IsBenefitSelectionAllowed	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> Indicates whether a member is allowed to select the benefit (<code>true</code>) or not (<code>false</code>). This field is available in API version 57.0 and later.  The default value is <code>false</code>.</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p>

Field	Details
	<p><b>Description</b> The most recent date on which a user viewed this record.</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Sort, Update</p> <p><b>Description</b> The name of the loyalty benefit.</p>
PrioritySequence	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> Priority rating of the loyalty benefit.</p>
UnitId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The unit of measure for quantity. This is a relationship field.</p> <p><b>Relationship Name</b> Unit</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> UnitOfMeasure</p>
Value	<p><b>Type</b> double</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The quantity included in the loyalty benefit.</p>

## BenefitParameterValue

---

Represents the runtime value of a parameter that's used when executing a flow for benefit action. This object is available in API version 52.0 and later.

### Supported Calls

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

### Fields

Field	Details
BenefitActionParameterId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The benefit action parameter value associated with the benefit action. This is a relationship field.</p> <p><b>Relationship Name</b> BenefitActionParameter</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> BenefitActionParameter</p>
BenefitId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> The benefit associated with the benefit action. This is a relationship field.</p> <p><b>Relationship Name</b> Benefit</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Benefit</p>

Field	Details
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Sort, Update</p> <p><b>Description</b> The name of the benefit parameter value.</p>
ParameterValue	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The pre-defined values of the parameter.</p>

## BenefitType

---

Represents information about the type of benefits such as Support Benefits available to the loyalty program member. This object is available in API version 51.0 and later.

### Supported Calls

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



## Fields

Field	Details
Category	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The category of the benefit type. Picklist values aren't provided for this field and must be added based on the requirements of the organization.</p>
Description	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The description of benefit type.</p>
ImageSource	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The source of the image that represents the benefit type. This field is available in API version 62.0 and later.</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
LoyaltyProgramId	<p><b>Type</b> reference</p>

Field	Details
	<p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> ID of the loyalty program. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgram</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgram</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Sort, Update</p> <p><b>Description</b> The name of the benefit type.</p>
OwnerId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> The ID of the user who owns this record. This is a polymorphic relationship field.</p> <p><b>Relationship Name</b> Owner</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Group, User</p>
ProcessType	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p><b>Description</b> Required. The process type associated with the loyalty benefit type. Possible values are:</p>

Field	Details
	<ul style="list-style-type: none"> <li>Loyalty</li> <li>Rebate</li> </ul>

## 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.

### [BenefitTypeChangeEvent on page 217 \(API Version 62.0\)](#)

Change events are available for the object.

### [BenefitTypeOwnerSharingRule](#)

Sharing rules are available for the object.

### [BenefitTypeShare](#)

Sharing is available for the object.

## GameDefinition

Represents information about a game that the loyalty program offers to its members. 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	<p><b>Type</b> textarea</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The description of the game.</p>
EndDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The date till which the game can be offered to members.</p>

Field	Details
ExpirationPeriod	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The number of days or months for which a game is valid for a member to play after the game is assigned to them.</p>
ExpirationPeriodUnit	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p><b>Description</b> The unit of measure for the ExpirationPeriod field.  Possible values are:</p> <ul style="list-style-type: none"> <li>• Days</li> <li>• Months</li> </ul>
IsGameAssignmentRequired	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> Indicates whether members are eligible to play the game only if they are assigned the game (<code>true</code>) or not (<code>false</code>).  The default value is <code>false</code>.</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>

Field	Details
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Sort, Update</p> <p><b>Description</b> The name of the game.</p>
OwnerId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> The owner of the record. This field is a polymorphic relationship field.</p> <p><b>Relationship Name</b> Owner</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Group, User</p>
StartDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Create, Filter, Sort, Update</p> <p><b>Description</b> The date from which the game can be offered to members.</p>
Status	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Restricted picklist, Sort, Update</p> <p><b>Description</b> The status of the game. Possible values are:</p> <ul style="list-style-type: none"> <li>• Active</li> <li>• Draft</li> <li>• Inactive</li> </ul>
TimeoutDuration	<p><b>Type</b> int</p>

Field	Details
	<p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The duration the game is valid for after a participant starts to play it. The value is specified in seconds.</p>
Type	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Restricted picklist, Sort, Update</p> <p><b>Description</b> The type of game. Possible values are:</p> <ul style="list-style-type: none"> <li>• Scratchcard</li> <li>• SpintheWheel—Spin The Wheel</li> </ul>

## 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.

### GameDefinitionFeed

Feed tracking is available for the object.

### GameDefinitionHistory

History is available for tracked fields of the object.

### GameDefinitionShare

Sharing is available for the object.

## GameParticipant

Represents a junction between a game definition and a participant who is either an account, a contact, or a loyalty program member. 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
AccountId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> ID of the participant's account that's associated with the game definition. This field is a relationship field.</p> <p><b>Relationship Name</b> Account</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Account</p>
ContactId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> ID of the participant's contact that's associated with the game definition. This field is a relationship field.</p> <p><b>Relationship Name</b> Contact</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Contact</p>
GameDefinitionId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> ID of the game definition that has been assigned to the participant. This field is a relationship field.</p> <p><b>Relationship Name</b> GameDefinition</p>

Field	Details
	<p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> GameDefinition</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
LoyaltyProgramMemberId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> ID of the participant's loyalty program member record that's associated with the game definition.  This field is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgramMember</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramMember</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p><b>Description</b> The autogenerated unique number of the record.</p>



Field	Details
OwnerId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> The owner of the record This field is a polymorphic relationship field.</p> <p><b>Relationship Name</b> Owner</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Group, User</p>
ParticipantId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> ID of the participant's contact, account, or loyalty program member record. This field is a polymorphic relationship field.</p> <p><b>Relationship Name</b> Participant</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Account, Contact, LoyaltyProgramMember</p>

## 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.

### [GameParticipantChangeEvent on page 217 \(API Version 63.0\)](#)

Change events are available for the object.

### [GameParticipantFeed](#)

Feed tracking is available for the object.

### [GameParticipantHistory](#)

History is available for tracked fields of the object.

### [GameParticipantShare](#)

Sharing is available for the object.

## GameParticipantReward

---

Represents the reward that's provided to a participant after they've played a game. 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
ExpirationDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The date till which the game is valid for the participant to play.</p>
GameParticipantId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> ID of the associated game participant. This field is a relationship field.</p> <p><b>Relationship Name</b> GameParticipant</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> GameParticipant</p>
GameRewardId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> ID of the reward offered to the participant. This field is a relationship field.</p> <p><b>Relationship Name</b> GameReward</p>

Field	Details
	<p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> GameReward</p>
IssuedRewardReferenceId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> ID of the voucher issued to the participant as a reward or the transaction journal that credits the points rewarded to the participant.  This field is a polymorphic relationship field.</p> <p><b>Relationship Name</b> IssuedRewardReference</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> TransactionJournal, Voucher</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p><b>Description</b> The autogenerated unique number of the record.</p>

Field	Details
SourceActivityId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> ID of the record that represents the reason why the participant was eligible to play the game. This field is a polymorphic relationship field.</p> <p><b>Relationship Name</b> SourceActivity</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> GameDefinition, Promotion, TransactionJournal</p>
Status	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Restricted picklist, Sort, Update</p> <p><b>Description</b> The participant's status for the game. Possible values are:</p> <ul style="list-style-type: none"> <li>• Expired</li> <li>• NoReward</li> <li>• Rewarded</li> <li>• YetToReward</li> </ul>

## 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.

### [GameParticipantRewardChangeEvent](#) on page 217 (API Version 63.0)

Change events are available for the object.

### [GameParticipantRewardFeed](#)

Feed tracking is available for the object.

### [GameParticipantRewardHistory](#)

History is available for tracked fields of the object.

# GameReward

---

The reward that's offered to participants of a game. 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
Color	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The color of the segment in the Spin the Wheel board that the represents the game reward.</p>
CurrentRewardCount	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The number of times the reward has been provided to game participants until the current date.</p>
Description	<p><b>Type</b> textarea</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The description of the reward.</p>
GameDefinitionId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> The game that the reward belongs to.  This field is a relationship field.</p>

Field	Details
	<p><b>Relationship Name</b> GameDefinition</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> GameDefinition</p>
ImageUrl	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The image URL of the segment in the Spin the Wheel board that the represents the game reward.</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
MaximumRewardCount	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The maximum number of times the reward can be provided to game participants.</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Sort, Update</p>

Field	Details
	<p><b>Description</b> The name of the reward.</p>
RewardDefinitionId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> ID of the voucher definition that's used to issue vouchers or the loyalty program currency that's used to credit points to participants as game reward.  This field is a polymorphic relationship field.</p> <p><b>Relationship Name</b> RewardDefinition</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramCurrency, VoucherDefinition</p>
RewardType	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Restricted picklist, Sort, Update</p> <p><b>Description</b> Possible values are:</p> <ul style="list-style-type: none"> <li>• CustomReward</li> <li>• LoyaltyPoints</li> <li>• NoReward</li> <li>• Raffle—Reserved for future use.</li> <li>• Voucher</li> </ul>
RewardValue	<p><b>Type</b> double</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The value of the currency or voucher that's issued to participants as game reward.</p>
WinProbability	<p><b>Type</b> double</p>

Field	Details
	<p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The probability of the reward being provided to a participant.</p>

## 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.

### [GameRewardChangeEvent](#) on page 217 (API Version 62.0)

Change events are available for the object.

### [GameRewardFeed](#)

Feed tracking is available for the object.

### [GameRewardHistory](#)

History is available for tracked fields of the object.

## JournalReason

Represents a view that specifies the reason for the action that created a transaction journal. This object is available in API version 51.0 and later.

 **Note:** 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

`describeSObjects()`, `query()`, `retrieve()`

## Fields

Field	Details
ApiName	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, idLookup, Sort</p> <p><b>Description</b> 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</p>



Field	Details
	<p>this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization.</p>
IsActive	<p><b>Type</b> picklist</p> <p><b>Properties</b> Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p><b>Description</b> Specifies if the Journal Reason picklist values in the Transaction Journal entity are active. Possible values are:</p> <ul style="list-style-type: none"> <li>• active</li> <li>• deleteInProgress</li> <li>• inactive</li> </ul>
JournalReasonType	<p><b>Type</b> picklist</p> <p><b>Properties</b> Filter, Group, Nillable, Restricted picklist, Sort</p> <p><b>Description</b> Provides the reason for the action that created a transaction journal. Possible values are:</p> <ul style="list-style-type: none"> <li>• AccrualReversal—Accrual Reversal</li> <li>• ManualPointsAdjustment—Manual Points Adjustment</li> <li>• RedemptionReversal—Redemption Reversal</li> </ul>
MasterLabel	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The label of the Journal Reason picklist.</p>

SEE ALSO:

[Salesforce Help: Prerequisites for Managing Member Transactions](#)

## JournalSubType

Represents the subcategory of transaction journal. This object is available in API version 51.0 and later.

## Supported Calls

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

## Fields

Field	Details
Description	<p><b>Type</b> textarea</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The description of journal subtype.</p>
JournalTypeId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> ID of the Journal Type. This is a relationship field.</p> <p><b>Relationship Name</b> JournalType</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> JournalType</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>

Field	Details
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Sort, Update</p> <p><b>Description</b> The name of the journal subtype.</p>

## 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.

**[JournalSubTypeChangeEvent](#) on page 217 (API Version 62.0)**

Change events are available for the object.

**[JournalSubTypeHistory](#)**

History is available for tracked fields of the object.

SEE ALSO:

[Salesforce Help: Prerequisites for Managing Member Transactions](#)

## JournalType

Represents the category of transactions. For example: accrual, redemption. This object is available in API version 51.0 and later.

## Supported Calls

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

## Fields

Field	Details
Description	<p><b>Type</b> textarea</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> Detailed description of the journal type.</p>

Field	Details
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Sort, Update</p> <p><b>Description</b> The name of the journal type.</p>
OwnerId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> The ID of the user who owns this record. This is a polymorphic relationship field.</p> <p><b>Relationship Name</b> Owner</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Group, User</p>
ProcessType	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p> <p><b>Description</b> Required. The process type associated with the journal type.</p>

Field	Details
	Possible value is: <ul style="list-style-type: none"> <li>Loyalty</li> </ul>

## 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.

### **JournalTypeChangeEvent on page 217 (API Version 62.0)**

Change events are available for the object.

### **JournalTypeHistory**

History is available for tracked fields of the object.

### **JournalTypeOwnerSharingRule**

Sharing rules are available for the object.

### **JournalTypeShare**

Sharing is available for the object.

SEE ALSO:

[Salesforce Help: Prerequisites for Managing Member Transactions](#)

## LoyaltyAggrPointExprLedger

Represents information about non-qualifying currency points that expire based on a fixed model tier group. This object is available in API version 54.0 and later.

## Supported Calls

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

LoyaltyAggrPointExprLedger isn't available in the Object Manager and can't be customized.

## Fields

Field	Details
ExpirationDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The date on which the points are set to expire.</p>

Field	Details
ExpirationPoints	<p><b>Type</b> double</p> <p><b>Properties</b> Create, Filter, Sort, Update</p> <p><b>Description</b> Required. The number of points that are set to expire on the expiration date.</p>
LoyaltyProgramCurrencyId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The loyalty program currency for which points are set to expire. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgramCurrency</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramCurrency</p>
LoyaltyProgramMemberId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The loyalty program member whose points are set to expire. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgramMember</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramMember</p>
PointsStatus	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p><b>Description</b> The status of the points that are set to expire.</p>

Field	Details
	<p>Possible values are:</p> <ul style="list-style-type: none"> <li>• Active</li> <li>• Inactive</li> </ul> <p>The default value is 'Active'.</p>
RedeemedPoints	<p><b>Type</b> double</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The number of points that have been redeemed by the associated loyalty program member out of the total points set to expire.</p>

## 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.

### [LoyaltyAggrPointExprLedgerChangeEvent](#)

Change events are available for the object.

## LoyaltyLedger

Records the points that have been credited or debited for a member across the transactions. This object is available in API version 51.0 and later.

## Supported Calls

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

## Fields

Field	Details
ActivityDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The date on which the transaction was performed by the loyalty program member.</p>

Field	Details
AdditionalNotes	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The notes about the transactions performed by the loyalty program members.</p>
Correlation	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The ID of any Salesforce or external record.  The field lets you store IDs that are up to 15 characters. If you store IDs that are fewer than 15 characters and use the field to query the object, add spaces after your query term to ensure that the query term is 15 characters.</p>
EscrowPointsCreditDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The date on which the escrow points are credited to the points balance for the loyalty ledger. This field is available in API version 53.0 and later.</p>
EventType	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p><b>Description</b> The type of the loyalty transaction.  Possible values are:</p> <ul style="list-style-type: none"> <li>• Credit</li> <li>• Debit</li> <li>• Expiry</li> <li>• Transfer</li> </ul> <p>The default value is 'Credit'.</p>
ExpiryDate	<p><b>Type</b> date</p>



Field	Details
	<p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The date on which the transaction expires.</p>
Flow	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The name of the flow.</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
LoyaltyPgmCrcySubtypeId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> Loyalty program currency subtype associated with the loyalty ledger. This field is available in API version 61.0 and later.  This field is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyPgmCrcySubtype</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyPgmCurrencySubtype</p>

Field	Details
LoyaltyProgramCurrencyId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> ID of the loyalty program currency. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgramCurrency</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramCurrency</p>
LoyaltyProgramMemberId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> ID of the loyalty program member. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgramMember</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramMember</p>
Points	<p><b>Type</b> double</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The points available for the loyalty program member.</p>
PromotionId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p>

Field	Details
	<p><b>Description</b> The promotion that's applied to the associated transaction resulting in the credit or debit of points. This field is available in API version 52.0 and later</p> <p>This is a relationship field.</p> <p><b>Relationship Name</b> Promotion</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Promotion</p>
TransactionJournalId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> ID of the transaction journal.</p> <p>This is a relationship field.</p> <p><b>Relationship Name</b> TransactionJournal</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> TransactionJournal</p>

## LoyaltyLedgerTraceability

Represents a tracker for the loyalty ledger. The debit ledger consumes points from the linked credit ledger for a member redemption or an accrual cancellation. This object is available in API version 62.0 and later.

### Supported Calls

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

LoyaltyAggrPointExprLedger isn't available in the Object Manager and can't be customized.

## Fields

Field	Details
ActionType	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Restricted picklist, Sort</p> <p><b>Description</b> Required. Specifies the action that creates an entry in the ledger. Possible values are:</p> <ul style="list-style-type: none"> <li>• CreditCancellation</li> <li>• CreditForArrears</li> <li>• Debit</li> <li>• DebitCancellation</li> <li>• DebitWithArrears</li> <li>• GroupTransfer</li> </ul> <p>The default value is Debit.</p>
CreditLoyaltyLedgerId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> Required. The credit type loyalty ledger associated with the loyalty ledger tracker. Points are consumed from the credit type ledger for the loyalty program member's debit type ledger. This field is a relationship field.</p> <p><b>Relationship Name</b> CreditLoyaltyLedger</p> <p><b>Refers To</b> LoyaltyLedger</p>
DebitLoyaltyLedgerId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> Required. The debit ledger associated with the loyalty ledger tracker. Points are consumed by the debit ledger from the loyalty program member's point balance. This field is a relationship field.</p> <p><b>Relationship Name</b> DebitLoyaltyLedger</p>

Field	Details
	<p><b>Refers To</b> LoyaltyLedger</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p><b>Description</b> The autogenerated number of the record.</p>
OwnerId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> The ID of the user who owns this record. This field is a polymorphic relationship field.</p> <p><b>Relationship Name</b> Owner</p> <p><b>Refers To</b> Group, User</p>
Points	<p><b>Type</b> double</p> <p><b>Properties</b> Create, Filter, Sort</p>

Field	Details
	<p><b>Description</b></p> <p>Required. The number of points used from the credit ledger by the debit ledger.</p>

## 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.

### [LoyaltyLedgerTraceabilityShare](#)

Sharing is available for the object.

## LoyaltyMemberCurrency

Represents the information about point balance for a member and a particular currency. This object is available in API version 51.0 and later.

## Supported Calls

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

## Fields

Field	Details
<code>BalanceBeforeReset</code>	<p><b>Type</b></p> <p>double</p> <p><b>Properties</b></p> <p>Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b></p> <p>The last balance of points before resetting the qualifying points of the member.</p>
<code>EscrowPointsBalance</code>	<p><b>Type</b></p> <p>double</p> <p><b>Properties</b></p> <p>Filter, Nillable, Sort</p> <p><b>Description</b></p> <p>The number of escrow points to be credited to the loyalty member's points balance. This field is available in API version 53.0 and later.</p> <p>This is a calculated field.</p>

Field	Details
ExpirablePoints	<p><b>Type</b> double</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> Calculates the points to be expired for unredeemed points.</p>
LastAccrualProcessedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The date when the last accrual was processed. This field indicates the date and time at which the Point Balance calculation process updated the total accrued points for a member and its currency the last time.</p>
LastEscrowProcessedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The last date and time when the Roll Over Escrow Points to Members Data Processing Engine definition was run to calculate rollover of escrow points. This field is available in API version 53.0 and later.</p>
LastExpirationProcessRunDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The most recent date when the expired points for a member were calculated.</p>
LastPointsAggregationDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The date on which the member's fixed type non-qualifying points were last aggregated.</p>
LastReferencedDate	<p><b>Type</b> dateTime</p>

Field	Details
	<p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastResetDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The most recent date when the qualifying points for members were reset.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
LoyaltyMemberId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> ID of the loyalty program member. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyMember</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramMember</p>
LoyaltyProgramCurrencyId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> ID of the loyalty program currency. This is a relationship field.</p>



Field	Details
	<p><b>Relationship Name</b> LoyaltyProgramCurrency</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramCurrency</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Sort, Update</p> <p><b>Description</b> The name of the loyalty member currency.</p>
NextResetDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The next date when the qualifying points for members are reset.</p>
PointsBalBeforeSecondLastReset	<p><b>Type</b> double</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The points balance before the second to last points reset of the member's associated qualifying currency. This field is available in API version 55.0 and later.</p>
PointsBalance	<p><b>Type</b> double</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> Represents information about Point balance for a member and a particular currency.</p>
RedemStlPendFromDateTime	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p>

Field	Details
	<p><b>Description</b> Date and time from when the member's redemption settlement is pending for the currency. This field is available in API version 61.0 and later.</p>
TotalEscrowPointsAccrued	<p><b>Type</b> double</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The number of escrow points that the associated member has accrued. This field is available in API version 53.0 and later.</p>
TotalEscrowRolloverPoints	<p><b>Type</b> double</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The total number of escrow points that are credited to the points balance until the current date. This field is available in API version 53.0 and later.</p>
TotalPointsAccrued	<p><b>Type</b> double</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The total points accrued by the member until today.</p>
TotalPointsExpired	<p><b>Type</b> double</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The total points expired for a member until today.</p>
TotalPointsRedeemed	<p><b>Type</b> double</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The total points redeemed by the member until today.</p>

## 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.

**LoyaltyMemberCurrencyChangeEvent** on page 217 (API Version 62.0)

Change events are available for the object.

**LoyaltyMemberCurrencyFeed** (API version 52.0)

Sharing rules are available for the object.

**LoyaltyMemberCurrencyHistory** (API version 52.0)

Sharing is available for the object.

## LoyaltyMembershipLifecycle

---

Represents information about a member's lifecycle in the loyalty program. This object is available in API version 57.0 and later.

## Supported Calls

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

## Fields

Field	Details
Comment	<p><b>Type</b> textarea</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The comments for the member's lifecycle event.</p>
EffectiveEndDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The date till when the member's lifecycle event is effective.</p>
ExternalIdentifier	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p>

Field	Details
	<p><b>Description</b> The external identifier of the member's lifecycle event.</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
LoyaltyProgramMemberId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> The loyalty program member whose lifecycle is tracked. This field is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgramMember</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramMember</p>
MembershipStatus	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. Specifies the status of the member's lifecycle event.</p>
MembershipType	<p><b>Type</b> picklist</p>

Field	Details
	<p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. Specifies the type of lifecycle event of the member.</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p><b>Description</b> Name of the lifecycle event.</p>
OrderItemId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The order associated with the lifecycle event. This field is a relationship field.</p> <p><b>Relationship Name</b> OrderItem</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> OrderItem</p>
EffectiveStartDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The date from when the member's lifecycle event is effective.</p>

## 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.

### [LoyaltyMembershipLifecycleHistory](#)

History is available for tracked fields of the object.

## LoyaltyMemberTier

---

Represents information about the current tier of the member across tier groups. This object is available in API version 51.0 and later.

### Supported Calls

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

### Fields

Field	Details
AreBenefitsAssigned	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> Indicates whether benefits are already assigned to a member tier.</p>
ChangeReasonType	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> Represents the information of the reason for changing a member tier. Possible values are:</p> <ul style="list-style-type: none"> <li>• Based on Past Activities or Tier</li> <li>• Resolve Tier Conflict</li> <li>• VIP Member</li> </ul>
EffectiveDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The date from which the tier is in effect.</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p>

Field	Details
	<p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
LoyaltyMemberId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> ID of the loyalty program member. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyMember</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramMember</p>
LoyaltyTierGroupId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> ID of the loyalty tier group. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyTierGroup</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyTierGroup</p>
LoyaltyTierId	<p><b>Type</b> reference</p>

Field	Details
	<p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> ID of the loyalty tier. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyTier</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyTier</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Sort, Update</p> <p><b>Description</b> The name of the loyalty member tier.</p>
ReasonForChange	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The reason to change a member tier.</p>
TierExpirationDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The date on which the tier expires.</p>

## 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.

**[LoyaltyMemberTierChangeEvent](#) on page 217 (API Version 62.0)**

Change events are available for the object.

**[LoyaltyMemberTierHistory](#) (API version 62.0)**

Sharing is available for the object.



## LoyaltyPartnerProduct

---

Represents products and product categories offered by loyalty partners. This object is available in API version 51.0 and later.

### Supported Calls

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

### Fields

Field	Details
EndDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The date on which a partner ends the offering of a product for accrual or redemption.</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
LoyaltyPartnerId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> ID of the loyalty program partner. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyPartner</p>

Field	Details
	<p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramPartner</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Sort, Update</p> <p><b>Description</b> The name of the loyalty partner product.</p>
PartnerCategoryDerivedFieldId	<p><b>Type</b> reference</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> A lookup field to show product category in Partner Product. This is a relationship field.</p> <p><b>Relationship Name</b> PartnerCategoryDerivedField</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> ProductCategory</p>
PartnerProductCost	<p><b>Type</b> currency</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The cost incurred by the partner when members redeem the product during the promotion. This field is available in API version 55.0 and later.</p>
ProductDerivedFieldId	<p><b>Type</b> reference</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> A lookup field to show product in Partner Product. This is a relationship field.</p>

Field	Details
	<p><b>Relationship Name</b> ProductDerivedField</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Product2</p>
ProductId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. ID of the partner product. This is a polymorphic relationship field.</p> <p><b>Relationship Name</b> Product</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Product2, ProductCategory</p>
StartDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The date on which a partner starts the offering of a product for accrual or redemption.</p>

## 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.

**[LoyaltyPartnerProductChangeEvent](#) on page 217 (API Version 52.0)**

Change events are available for the object.

**[LoyaltyPartnerProductHistory](#) (API version 52.0)**

History is available for tracked fields of the object.

## LoyaltyPgmCurrencySubtype

---

Represents a subtype of a fixed-model non-qualifying loyalty program currency. 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

This object is available only if the Loyalty Management - Advanced license is enabled in your org.

### Fields

Field	Details
CostPerUnit	<p><b>Type</b> currency</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> Monetary value of each unit of the loyalty program currency subtype.</p>
Description	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> Description of the loyalty program currency subtype.</p>
IsActive	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. Indicates whether the loyalty program currency subtype is active (true) or not (false). The default value is <code>false</code>.</p>
LastReferencedDate	<p><b>Type</b> dateTime</p>

Field	Details
	<p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> Most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> Most recent date on which a user viewed this record.</p>
LoyaltyProgramCurrencyId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> Required. The fixed-model non-qualifying loyalty program currency associated with the loyalty program currency subtype. This field is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgramCurrency</p> <p><b>Relationship Type</b> Master-detail</p> <p><b>Refers To</b> LoyaltyProgramCurrency (the master object)</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Sort, Update</p> <p><b>Description</b> Required. The name of the loyalty program currency subtype.</p>
UsagePriorityNumber	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Sort, Update</p>

Field	Details
	<p><b>Description</b></p> <p>Required. The usage priority number of the loyalty program currency subtype. The usage priority number decides the order in which the points are redeemed across currency subtypes when multiple currency subtypes expire on the same date.</p>

## LoyaltyPgmEngmtAttribute

Represents information about an engagement attribute for a loyalty program. This object is available in API version 55.0 and later.

### Supported Calls

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

### Fields

Field	Details
DefaultValue	<p><b>Type</b></p> <p>string</p> <p><b>Properties</b></p> <p>Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b></p> <p>The default value for the loyalty program engagement attribute.</p>
Description	<p><b>Type</b></p> <p>textarea</p> <p><b>Properties</b></p> <p>Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b></p> <p>The description for the loyalty program engagement attribute.</p>
EndDate	<p><b>Type</b></p> <p>date</p> <p><b>Properties</b></p> <p>Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b></p> <p>The date when the loyalty program engagement attribute ends.</p>
LastReferencedDate	<p><b>Type</b></p> <p>dateTime</p>

Field	Details
	<p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
LoyaltyProgramId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> The loyalty program that's related to the loyalty program engagement attribute. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgram</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgram</p>
MaximumResetCountPerPeriod	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The number of times the current value of an engagement attribute can be reset during a period.</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Sort, Update</p> <p><b>Description</b> The name of the loyalty program engagement attribute.</p>

Field	Details
ResetValueOn	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The day on which loyalty program members' current value for the engagement attribute must be reset.</p> <p>If the reset frequency is Weekly, specify the day of the week when the current value is reset. To reset the current value on a Sunday, enter 1. To reset the current value on a Saturday, enter 7. If the reset frequency is Monthly, enter a value between 1 and 31. To reset the current value on the fifteenth of every month, enter 15. If the reset frequency is Yearly, enter a value between 1 and 12. To reset the current value on the first day of April month, enter 4. If the reset frequency is custom, enter the number of days after which the current value is reset.</p>
ShouldOverwriteAttrCurrVal	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> Indicates whether the associated member's current value is overwritten every period (<code>true</code>) or not (<code>false</code>).</p> <p>The default value is <code>false</code>.</p>
StartDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The date when the loyalty program engagement attribute ends.</p>
Status	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p> <p><b>Description</b> Required. The status of the loyalty program engagement attribute.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>• Active</li> <li>• Draft</li> <li>• Inactive</li> </ul> <p>The default value is <code>Draft</code>.</p>



Field	Details
TargetValue	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The value that the loyalty program member must reach to qualify for rewards of the engagement attribute.</p>
ValueResetFrequency	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p><b>Description</b> The frequency at which the loyalty program member's engagement attribute value is reset. Possible values are:</p> <ul style="list-style-type: none"> <li>• CustomPeriod—Custom Period</li> <li>• Daily</li> <li>• Monthly</li> <li>• Weekly</li> <li>• Yearly</li> </ul>

## 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.

### [LoyaltyPgmEngmtAttributeChangeEvent](#) on page 217 (API Version 62.0)

Change events are available for the object.

### [LoyaltyPgmEngmtAttributeHistory](#)

History is available for tracked fields of the object.

## LoyaltyPgmEngmtAttrProm

Represents a junction between Loyalty Program Engagement Attribute and Promotion. Associate an engagement attribute with a promotion if you want to track the engagement attribute during the promotion period. This object is available in API version 55.0 and later.

## Supported Calls

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

## Fields

Field	Details
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
LoyaltyProgramEngmtAttrId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> The loyalty program engagement attribute associated with the promotion. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgramEngmtAttr</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyPgmEngmtAttribute</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p><b>Description</b> The name of the loyalty program engagement attribute promotion. The name is autopopulated.</p>
PromotionId	<p><b>Type</b> reference</p>

Field	Details
	<p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> The promotion associated with the loyalty program engagement attribute. This is a relationship field.</p> <p><b>Relationship Name</b> Promotion</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Promotion</p>

## 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.

**[LoyaltyPgmEngmtAttrPromChangeEvent](#) on page 217 (API Version 62.0)**

Change events are available for the object.

**[LoyaltyPgmEngmtAttrPromHistory](#)**

History is available for tracked fields of the object.

## LoyaltyPgmGroupMbrRlnsp

Represents a junction between a Corporate or Individual type loyalty program member who is part of a group and the Group type loyalty program member. This object is available in API version 53.0 and later.

## Supported Calls

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

## Fields

Field	Details
CanMemberRedeemFromGroup	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p>

Field	Details
	<p><b>Description</b> Required. Indicates whether the member can redeem points from a group. This field isn't used. You can customize and use this field based on your org's requirements. The default value is 'false'.</p>
ContributionCalculationType	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Restricted picklist, Sort, Update</p> <p><b>Description</b> Required. Specifies the basis on which the member's point contribution to a group is calculated. Possible values are:</p> <ul style="list-style-type: none"> <li>• Percentage</li> </ul>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
LoyaltyProgramGroupMemberId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> ID of the Group type loyalty program member that the related Individual or Corporate type loyalty program member is a part of. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgramGroupMember</p> <p><b>Relationship Type</b> Lookup</p>

Field	Details
	<p><b>Refers To</b> LoyaltyProgramMember</p>
MemberName	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The name of the related loyalty program member record. For an Individual type loyalty program member, it's the first name and the last name of the contact record associated with loyalty program member record. For a Corporate type loyalty program member, it's the name of the account record associated with the loyalty program member record. This field is read-only.</p>
MemberPointContributionPercent	<p><b>Type</b> percent</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The percentage of a group member's total accrued points that's transferred to the group.</p>
MemberRole	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> Specifies the role of the loyalty program member in a group. You can add values to the picklist. This field isn't used. You can customize and use this field based on your org's requirements.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>• Primary</li> <li>• Secondary</li> </ul>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, idLookup, Sort</p> <p><b>Description</b> The name of the loyalty program member group relationship record.</p>
RelatedLoyaltyProgramMemberId	<p><b>Type</b> reference</p>

Field	Details
	<p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> The related Individual or Corporate type loyalty program member associated with the loyalty program group member relationship.  This is a relationship field.</p> <p><b>Relationship Name</b> RelatedLoyaltyProgramMember</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramMember</p>

## 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.

### [LoyaltyPgmGroupMbrRInspChangeEvent](#) on page 217 (API Version 62.0)

Change events are available for the object.

### [LoyaltyPgmGroupMbrRInspHistory](#)

History is available for tracked fields of the object.

## LoyaltyPgmMbrAttributeVal

Represents information about the value reached by a loyalty program member for an engagement attribute. This object is available in API version 55.0 and later.

## Supported Calls

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

## Fields

Field	Details
CumulativeValue	<p><b>Type</b> double</p> <p><b>Properties</b> Create, Filter, Nillable, Sort</p>

Field	Details
	<p><b>Description</b> The cumulative value achieved by a loyalty program member for the related loyalty program engagement attribute across all the periods.</p>
CurrentValue	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> The value achieved by a loyalty program member for the related loyalty program member engagement attribute in during a specific period.</p>
EndDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> The end date of the period for which the member's progress towards achieving the related engagement attribute target is tracked.</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastResetDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> The date on which the related loyalty program member's current value was last reset.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>

Field	Details
LoyaltyPgmEngmtAttributeId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> Required. The loyalty program engagement attribute that's related to the loyalty program member engagement attribute value.  This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyPgmEngmtAttribute</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyPgmEngmtAttribute</p>
LoyaltyProgramMemberId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> The loyalty program member that's related to the loyalty program member engagement attribute value.  This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgramMember</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramMember</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p><b>Description</b> The name of the loyalty program engagements attribute related to the loyalty program member engagement attribute value.</p>
ResetCount	<p><b>Type</b> int</p>



Field	Details
	<p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> The number of times the current value of the engagement attribute is reset.</p>
StartDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> Required. The start date of the period for which the member's progress towards achieving the related engagement attribute target is tracked.</p>

## 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.

**[LoyaltyPgmMbrAttributeValChangeEvent](#) on page 217 (API Version 62.0)**

Change events are available for the object.

**[LoyaltyPgmMbrAttributeValHistory](#)**

History is available for tracked fields of the object.

## LoyaltyPgmMbrPromEligView

Represents a virtual object that provides a consolidated view of the loyalty program member's eligibility for a promotion and their enrollment status in eligible promotions. This object is available in API version 56.0 and later.

## Supported Calls

`create()`, `describeSObjects()`, `query()`

LoyaltyPgmMbrPromEligView isn't available in the Object Manager and can't be customized.

## Fields

Field	Details
CurrPromotionLiabilityAmt	<p><b>Type</b> currency</p> <p><b>Properties</b> Create, Filter, Nillable, Sort</p>

Field	Details
	<p><b>Description</b> The current liability amount of the related promotion.</p>
CurrencyIsoCode	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Restricted picklist, Sort</p> <p><b>Description</b> The currency ISO code of the promotion. This field is only available in orgs where multiple currencies are enabled.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>• USD—U.S. Dollar</li> </ul> <p>The default value is USD.</p>
CurrentPromotionPointsCount	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> The liability amount incurred for the related promotion till date.</p>
DefaultVoucherDefinitionId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> The voucher definition that's used as the default in the related promotion.</p> <p>This field is a relationship field.</p> <p><b>Relationship Name</b> DefaultVoucherDefinition</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> VoucherDefinition</p>
Description	<p><b>Type</b> textarea</p> <p><b>Properties</b> Create, Filter, Nillable, Sort</p> <p><b>Description</b> The description of the LoyaltyPgmMbrPromEligView record.</p>

Field	Details
DoesPromHavePtWithFctr	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort</p> <p><b>Description</b> Indicates whether the related promotion has a point with factor. The default value is <code>false</code>.</p>
EndDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> The end date of the related promotion.</p>
EndTime	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Create, Filter, Nillable, Sort</p> <p><b>Description</b> End date and time of the promotion.</p>
ExternalId	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> A combination of the related loyalty program member ID and the related promotion ID.</p>
FulfillmentAction	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p><b>Description</b> Specifies the fulfillment action of the related promotion. Possible values are:</p> <ul style="list-style-type: none"> <li>• <code>CreditPoints</code></li> <li>• <code>IssueVoucher</code></li> </ul>
IsMbrEligBsdCmpnOrSgmt	<p><b>Type</b> boolean</p>

Field	Details
	<p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort</p> <p><b>Description</b> Indicates if the member who is eligible for a promotion is part of a campaign or segment. The default value is <code>false</code>.</p>
IsPromotionEnrollmentRqr	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort</p> <p><b>Description</b> Indicates whether the related promotion requires member enrollment. The default value is <code>false</code>.</p>
IssuedPromotionVoucherCount	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> The number of vouchers issued for the related promotion.</p>
LoyaltyProgramCurrencyId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> The loyalty program currency that's used to credit points for the related promotion. This field is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgramCurrency</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramCurrency</p>
LoyaltyProgramId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> The loyalty program that's associated with the related promotion.</p>

Field	Details
	<p>This field is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgram</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgram</p>
LoyaltyProgramMemberId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> The loyalty program member associated with the related promotion. This field is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgramMember</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramMember</p>
LoyaltyPromotionType	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p><b>Description</b> Specifies the type of the related promotion. Possible values are:</p> <ul style="list-style-type: none"> <li>• Cumulative</li> <li>• Joint</li> <li>• Standard</li> </ul>
MaxPromotionLiabilityAmount	<p><b>Type</b> currency</p> <p><b>Properties</b> Create, Filter, Nillable, Sort</p> <p><b>Description</b> The maximum liability amount that the related promotion can incur.</p>

Field	Details
MaxPromotionVoucherCount	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> The maximum vouchers that can be issued for the related promotion.</p>
MaximumPromotionRewardValue	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> The maximum reward value of the related promotion.</p>
MemberEligibilityCategory	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p><b>Description</b> Specifies whether the loyalty program member is eligible for the promotion. Possible values are:</p> <ul style="list-style-type: none"> <li>• Eligible</li> <li>• EligibleButNotEnrolled</li> <li>• Ineligible</li> </ul>
Objective	<p><b>Type</b> textarea</p> <p><b>Properties</b> Create, Filter, Nillable, Sort</p> <p><b>Description</b> The objective of the related promotion.</p>
PromEnrollmentStartDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> The date when the enrollment in the related promotion starts.</p>
PromotionCumulativeUsageTgt	<p><b>Type</b> int</p>

Field	Details
	<p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> The cumulative usage target of the related promotion.</p>
PromotionEnrollmentEndDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> The date when the enrollment in the related promotion ends.</p>
PromotionId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> The promotion related to the promotion related to the LoyaltyPgmMbrPromEligView record. This field is a relationship field.</p> <p><b>Relationship Name</b> Promotion</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Promotion</p>
PromotionImageUrl	<p><b>Type</b> url</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> The URL of the promotion's image location. This field is available in API version 58.0 and later.</p>
PromotionName	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> The name of the related promotion.</p>

Field	Details
PromotionPointFactor	<p><b>Type</b> double</p> <p><b>Properties</b> Create, Filter, Nillable, Sort</p> <p><b>Description</b> The points factor of the related promotion.</p>
PromotionTrxnJournalType	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p><b>Description</b> The transaction journal type that's associated with the related promotion.  Possible values are:</p> <ul style="list-style-type: none"> <li>• Accrual</li> <li>• Redemption</li> </ul>
StartDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> Required. The start date of the related promotion.</p>
StartDateTime	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Create, Filter, Nillable, Sort</p> <p><b>Description</b> The start date and time of the promotion.</p>
TotalPromotionRewardPointsVal	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> The total points credited to the loyalty program member for the related promotion.</p>



# LoyaltyPgmPartnerCurrency

---

Represents a junction between a loyalty program partner and a loyalty program currency. This object is available in API version 55.0 and later.

## Supported Calls

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

## Fields

Field	Details
AccrualConversionFactor	<p><b>Type</b> double</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The factor by which points earned by a loyalty program member for an accrual-type transaction are converted to determine the number of points to be debited from the associated loyalty program partner's points balance or balance amount.</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
LoyaltyProgramCurrencyId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> Required. The loyalty program currency associated with the loyalty program partner currency record.</p>

Field	Details
	<p>This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgramCurrency</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramCurrency</p>
LoyaltyProgramPartnerId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> The loyalty program partner associated with the loyalty program partner currency record. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgramPartner</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramPartner</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p><b>Description</b> The name of the loyalty program partner currency.</p>
RedemptionConversionFactor	<p><b>Type</b> double</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The factor by which points redeemed by a loyalty program member for a redemption-type transaction are converted to determine the number of points or amount to be credited to the associated loyalty program partner's points balance or balance amount.</p>

## 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.

### LoyaltyPgmPartnerCurrencyHistory

History is available for tracked fields of the object.

## LoyaltyPgmPartnerPromotion

---

Represents a junction between the Promotion and Loyalty Program Partner objects. This object is available in API version 55.0 and later.

## Supported Calls

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

## Fields

Field	Details
ContributionPercentage	<p><b>Type</b> percent</p> <p><b>Properties</b> Create, Filter, Sort, Update</p> <p><b>Description</b> Required. The percentage of contribution given by a partner to the promotion.</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
LoyaltyProgramPartnerId	<p><b>Type</b> reference</p>

Field	Details
	<p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The loyalty program partner associated with the loyalty program partner prepaid pack or postpaid balance amount.  This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgramPartner</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramPartner</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p><b>Description</b> The name of the loyalty program partner promotion.</p>
PromotionId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> The joint promotion in which the loyalty program partner is a part of.  This is a relationship field.</p> <p><b>Relationship Name</b> Promotion</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Promotion</p>

## 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.

**[LoyaltyPgmPartnerPromotionChangeEvent](#) on page 217 (API Version 62.0)**

Change events are available for the object.

**LoyaltyPgmPartnerPromotionHistory**

History is available for tracked fields of the object.

## LoyaltyPgmPtnrLdgrSummary

---

Represents the aggregated ledger used by a loyalty program to credit points to or debit points from a prepaid partner's points pack. This ledger can also be used to add or deduct the monetary value of points from the postpaid partner's bill. This object is available in API version 55.0 and later.

### Supported Calls

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

### Fields

Field	Details
ActivityDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The date on which points that are part of the associated loyalty program partner prepaid pack are debited or credited. For postpaid billing, it is the date when the postpaid amount of the loyalty program partner is debited or credited.</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
LoyaltyPgmPtnrPrepaidPackId	<p><b>Type</b> reference</p>

Field	Details
	<p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The loyalty program partner prepaid pack associated with the loyalty program partner ledger. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyPgmPtnrPrepaidPack</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyPgmPtnrPrepaidPack</p>
LoyaltyProgramPartnerId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> The loyalty program partner associated with the loyalty program. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgramPartner</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramPartner</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p><b>Description</b> The name of the loyalty program partner ledger summary.</p>
Status	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p> <p><b>Description</b> Required. Specifies the status of the loyalty program partner ledger. Possible values are:</p>

Field	Details
	<ul style="list-style-type: none"> <li>• <b>Pending</b> Partner doesn't have the adequate balance and hence the ledger is pending to be processed.</li> <li>• <b>Processed</b> Partner's ledger is processed.</li> </ul> <p>The default value is <code>Pending</code>.</p>
<code>TotalPoints</code>	<p><b>Type</b> double</p> <p><b>Properties</b> Create, Filter, Sort, Update</p> <p><b>Description</b> Required. The number of points debited from or credited to the associated loyalty program partner's prepaid pack or postpaid bill amount.</p>
<code>Type</code>	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p> <p><b>Description</b> Required. Specifies the type of the loyalty program partner ledger. Possible values are:</p> <ul style="list-style-type: none"> <li>• <code>Credit</code></li> <li>• <code>CreditCancellation</code>—Credit Cancellation</li> <li>• <code>Debit</code></li> <li>• <code>DebitCancellation</code>—Debit Cancellation</li> </ul> <p>The default value is <code>Credit</code>.</p>

## LoyaltyPgmPtnrPrepaidPack

Represents information about a prepaid set of points purchased by a loyalty program partner. This object is available in API version 55.0 and later.

### Supported Calls

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

## Fields

Field	Details
EndDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The date until when the loyalty program partner prepaid pack is effective.</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
LoyaltyProgramPartnerId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> The loyalty program partner associated with the loyalty program partner prepaid pack. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgramPartner</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramPartner</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Sort, Update</p>



Field	Details
	<p><b>Description</b> The name of the prepaid points pack.</p>
OrderId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The order associated with the loyalty program partner prepaid pack. This is a relationship field.</p> <p><b>Relationship Name</b> Order</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Order</p>
PurchaseDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> Required. The date when the prepaid pack was purchased by the loyalty program partner.</p>
PurchasedPoints	<p><b>Type</b> double</p> <p><b>Properties</b> Create, Filter, Sort</p> <p><b>Description</b> Required. The number of points credited to the associated loyalty program partner when a prepaid pack of points is purchased.</p>
RemainingPoints	<p><b>Type</b> double</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The number of points remaining in the loyalty program partner's prepaid pack.</p>

## 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.

### LoyaltyPgmPtrPrepaidPackHistory

History is available for tracked fields of the object.

## LoyaltyProgram

---

Represents the loyalty program within an organization. All the information related to members, tiers, tier groups, program currencies, vouchers, benefits, partners, promotions, transactions are associated with the program. This object is available in API version 51.0 and later.

## Supported Calls

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

## Fields

Field	Details
Description	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> Description of the loyalty program.</p>
EscrowPeriod	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The number of days after which the escrow points are credited to the existing points balance. This field is available in API version 53.0 and later.</p>
IsPrimary	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> Indicates whether a loyalty program is the primary program. The default value is 'false'. This field is available in API version 52.0 and later.</p>

Field	Details
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Sort, Update</p> <p><b>Description</b> Required. The name of the loyalty program. The name must contain at least one alphanumeric character.</p>
OwnerId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> ID of the owner of this record.  This is a polymorphic relationship field.</p> <p><b>Relationship Name</b> Owner</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Group, User</p>
Status	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p>

Field	Details
	<p><b>Description</b></p> <p>Required. The status of the loyalty program.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>• Active</li> <li>• Inactive</li> </ul>

## 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.

### [LoyaltyProgramChangeEvent](#) on page 217 (API Version 62.0)

Change events are available for the object.

### [LoyaltyProgramFeed](#)

Feed tracking is available for the object.

### [LoyaltyProgramHistory](#)

History is available for tracked fields of the object.

### [LoyaltyProgramOwnerSharingRule](#)

Sharing rules are available for the object.

### [LoyaltyProgramShare](#)

Sharing is available for the object.

## LoyaltyProgramBadge

Represents information about the badge that's associated with a loyalty program. This object is available in API version 56.0 and later.

## Supported Calls

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

## Fields

Field	Details
Description	<p><b>Type</b></p> <p>string</p> <p><b>Properties</b></p> <p>Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b></p> <p>The description of the loyalty program badge.</p>

Field	Details
ImageUrl	<p><b>Type</b> url</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The URL of the image location for the loyalty program badge.</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
LoyaltyProgramId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> The loyalty program that's associated with the badge. This field is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgram</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgram</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Sort, Update</p> <p><b>Description</b> The name of the loyalty program badge.</p>

Field	Details
StartDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The date from when the loyalty program badge can be assigned to members.</p>
Status	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. Specifies the status of the loyalty program badge. Possible values are:</p> <ul style="list-style-type: none"> <li>• Active</li> <li>• Draft</li> <li>• Inactive</li> </ul>
ValidityDuration	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> Specifies the unit of measure for the validity period.</p>
ValidityDurationUnit	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p><b>Description</b> The number of days or months in the validity period. Possible values are:</p> <ul style="list-style-type: none"> <li>• Days</li> <li>• Months</li> </ul>
ValidityEndDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p>

Field	Details
	<p><b>Description</b></p> <p>The date after which the badge is no longer assigned to members. Select a value for this field if the badge's validity type is <code>FixedDate</code>.</p>
ValidityType	<p><b>Type</b></p> <p>picklist</p> <p><b>Properties</b></p> <p>Create, Filter, Group, Restricted picklist, Sort, Update</p> <p><b>Description</b></p> <p>Required. Specifies the model used to determine the duration for which the badge is valid for a member.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>• <code>FixedDate</code>—Fixed Date</li> <li>• <code>Period</code></li> </ul>

## 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.

### [LoyaltyProgramBadgeChangeEvent](#) on page 217 (API Version 62.0)

Change events are available for the object.

### [LoyaltyProgramBadgeHistory](#)

History is available for tracked fields of the object.

## LoyaltyProgramCurrency

Represents qualifying and non-qualifying currencies associated with the program. Qualifying currency represents the engagement of a member with the program and it's associated with a tier group. Non-qualifying currency represent the points that the member earns for redemption. This object is available in API version 51.0 and later.

## Supported Calls

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

## Fields

Field	Details
CostperUnit	<p><b>Type</b></p> <p>currency</p>

Field	Details
	<p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The cost for each currency associated with a loyalty program.</p>
CurrencyType	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Restricted picklist, Sort</p> <p><b>Description</b> Required. Type of the loyalty program currency. Possible values are:</p> <ul style="list-style-type: none"> <li>• NonQualifying</li> <li>• Qualifying</li> </ul>
DoesAllowLedgerTracing	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> Indicates whether points credited by accrual transactions and debited by redemption transactions for a fixed-model non-qualifying currency is traced (true) or not (false). This field is available in API version 61.0 and later.  The default value is <code>false</code>.</p>
DoesAllowNegativeBalance	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> Indicates whether a non-qualifying currency allows members to hold a negative point balance. This field is available in API version 61.0 and later.  The default value is <code>false</code>.</p>
ExpiryModel	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p><b>Description</b> The model evaluates the type of expiration of points based on the fixed date or customer's activity.</p>



Field	Details
	<p>Possible values are:</p> <ul style="list-style-type: none"> <li>• Activity</li> <li>• Fixed</li> </ul>
ExpiryPeriodFrequency	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The frequency type to evaluate the expiration of currency.</p>
ExpiryPeriodUnit	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p><b>Description</b> Unit of the period in which currency expiration is done.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>• Days</li> <li>• Months</li> <li>• Years</li> </ul>
ExtendExpiration	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p><b>Description</b> The field decides if the expiration date is changed on the exact day or the last day of the month.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>• None</li> <li>• TillEndOfMonth</li> </ul>
HasEscrow	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> Indicates whether points can be escrowed for the loyalty program currency. This field is available in API version 53.0 and later.</p>

Field	Details
	The default value is <code>false</code> .
<code>IsActive</code>	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. Indicates whether the loyalty program currency is active.</p>
<code>IsPrimary</code>	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> Indicates whether the loyalty program currency is primary.</p>
<code>LastReferencedDate</code>	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
<code>LastViewedDate</code>	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
<code>LoyaltyProgramId</code>	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> The ID of the loyalty program. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgram</p> <p><b>Relationship Type</b> Lookup</p>

Field	Details
	<p><b>Refers To</b> LoyaltyProgram</p>
LoyaltyTierGroupId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The ID of the loyalty tier group. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyTierGroup</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyTierGroup</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Sort, Update</p> <p><b>Description</b> The name of the loyalty program currency.</p>

## 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.

**LoyaltyProgramCurrencyChangeEvent on page 217 (API Version 62.0)**

Change events are available for the object.

## LoyaltyProgramCurrencyTier

Represents a junction between non-qualifying loyalty program currency and loyalty tier. 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

This object is available only if the Loyalty Management - Growth or the Loyalty Management - Advanced license is enabled in your org.

## Fields

Field	Details
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> Most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> Most recent date on which a user viewed this record.</p>
LoyaltyProgramCurrencyId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> Non-qualifying loyalty program currency associated with the loyalty tier. This field is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgramCurrency</p> <p><b>Relationship Type</b> Master-detail</p> <p><b>Refers To</b> LoyaltyProgramCurrency (the master object)</p>
LoyaltyTierId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> Loyalty tier associated with the loyalty program currency. This field is a relationship field.</p>

Field	Details
	<p><b>Relationship Name</b> LoyaltyTier</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyTier</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p><b>Description</b> Autogenerated number of the loyalty program currency tier.</p>
NegativePointsBalanceLimit	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> Maximum negative point balance that members of the loyalty tier can hold for the associated loyalty program currency.</p>

## LoyaltyProgramMbrPromotion

Represents details about the promotions to be applied for a loyalty program member. Tracks the enrollment status, promotion status, and progress for a promotion of a member. This object is available in API version 51.0 and later.

### Supported Calls

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

### Fields

Field	Details
CumulativeUsageCompletePercent	<p><b>Type</b> percent</p> <p><b>Properties</b> Filter, Nillable, Sort</p>

Field	Details
	<p><b>Description</b> The percentage of the promotion completed by the member.</p>
CumulativeUsageCompleted	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The amount of the frequency type promotion completed by the member.</p>
CumulativeUsageTarget	<p><b>Type</b> int</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The target of the frequency type promotion that must be achieved by the member to complete the promotion.</p>
IsEnrollmentActive	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> Indicates whether the enrollment for the promotion is active.</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
LoyaltyProgramMemberId	<p><b>Type</b> reference</p>

Field	Details
	<p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> The loyalty program member associated with the promotion. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgramMember</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramMember</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p><b>Description</b> Unique identifier for the loyalty program member promotion.</p>
PromotionId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> The ID of the loyalty program member promotion. This is a relationship field.</p> <p><b>Relationship Name</b> Promotion</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Promotion</p>

## 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.

### [LoyaltyProgramMbrPromotionChangeEvent](#)

Change events are available for the object.

**LoyaltyProgramMbrPromotionHistory**

History is available for tracked fields of the object.

## LoyaltyProgramMember

---

Represents information about member who belongs to a loyalty program. The member can be individual or corporate type. This object is available in API version 51.0 and later.

### Supported Calls

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

### Fields

Field	Details
AccountId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> ID of the account with which the loyalty program member is associated. This field is optional for a Group type member.  This is a relationship field.</p> <p><b>Relationship Name</b> Account</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Account</p>
CanReceivePartnerPromotions	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> Indicates whether the member can receive partner promotions for the program (true) or not (false).</p>
CanReceivePromotions	<p><b>Type</b> boolean</p>



Field	Details
	<p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> Indicates whether the member can receive promotions for the program (<code>true</code>) or not (<code>false</code>).</p>
ContactId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> ID of the contact with which the loyalty program member is associated. This field is optional for a Group type member.  This is a relationship field.</p> <p><b>Relationship Name</b> Contact</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Contact</p>
EnrollmentChannel	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p><b>Description</b> Channel through which membership is enrolled.  Possible values are:</p> <ul style="list-style-type: none"> <li>• CallCenter</li> <li>• Email</li> <li>• Franchise</li> <li>• Mobile</li> <li>• Partner</li> <li>• POS</li> <li>• Print</li> <li>• Social</li> <li>• Store</li> <li>• Web</li> </ul>

Field	Details
EnrollmentDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. Date when the member enrolled to the loyalty program.</p>
GroupCreatedByMemberId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> The Individual or Corporate type loyalty program member who created the group or requested for the group to be created. This field is available in API version 53.0 and later.  This is a relationship field.</p> <p><b>Relationship Name</b> GroupCreatedByMember</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramMember</p>
GroupName	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The name of the group when the type of membership in the loyalty program is Group. This field is available in API version 53.0 and later.</p>
IsEligibleForTierAssessment	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> Indicates whether the member is eligible for tier assessment. A member is eligible for tier assessment when there is a change in their qualifying points balance. This field is available in API version 58.0 and later.  The default value is <code>false</code>.</p>

Field	Details
LastActivityDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The date when the loyalty program member last performed a transaction.</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
MemberStatus	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The status of the member in the loyalty program. Possible values are:</p> <ul style="list-style-type: none"> <li>• Active</li> <li>• Inactive</li> <li>• Merged—This value is available in API version 56.0 and later.</li> </ul>
MemberType	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Restricted picklist, Sort</p> <p><b>Description</b> Required. The type of membership in the loyalty program. Possible values are:</p> <ul style="list-style-type: none"> <li>• Corporate</li> </ul>

Field	Details
	<ul style="list-style-type: none"> <li>• Group</li> <li>• Individual</li> </ul>
MembershipEndDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The date on which the membership ends.</p>
MembershipLastRenewalDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The most recent date when the membership was renewed.</p>
MembershipNumber	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Sort, Update</p> <p><b>Description</b> The membership number of the loyalty program.</p>
OwnerId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> The ID of the user who owns this record. This is a polymorphic relationship field.</p> <p><b>Relationship Name</b> Owner</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Group, User</p>
ProgramId	<p><b>Type</b> reference</p>

Field	Details
	<p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> ID of the loyalty program. This is a relationship field.</p> <p><b>Relationship Name</b> Program</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgram</p>
ReferralCode	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Nillable, Sort, Update</p> <p><b>Description</b> The code that the member can share with others who want to join the loyalty program. If no referral code is entered during enrollment, an automatic referral code is generated for the member. This field is available in API version 57.0 and later.</p>
ReferredById	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> Reference to Member who referred to the program. This is a relationship field.</p> <p><b>Relationship Name</b> ReferredBy</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramMember</p>
RelatedCorporateMembershipId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The related corporate membership ID of the loyalty program.</p>

Field	Details
	<p>This is a relationship field.</p> <p><b>Relationship Name</b> RelatedCorporateMembership</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramMember</p>
StatementFrequency	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p><b>Description</b> The frequency of loyalty balance statements.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>• Monthly</li> <li>• Quarterly</li> </ul>
StatementLastGeneratedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The date on which the statement is last generated.</p>
StatementMethod	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p><b>Description</b> The mode of loyalty balance statements.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>• Email</li> <li>• Mail</li> </ul>

## 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.

**LoyaltyProgramMemberChangeEvent**

Change events are available for the object.

**LoyaltyProgramMemberHistory**

History is available for tracked fields of the object.

**LoyaltyProgramMemberOwnerSharingRule**

Sharing rules are available for the object.

**LoyaltyProgramMemberShare**

Sharing is available for the object.

## LoyaltyProgramMemberBadge

---

Represents a badge that's assigned to a loyalty program member. This object is available in API version 56.0 and later.

### Supported Calls

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

### Fields

Field	Details
EndDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The date until when the loyalty program badge is assigned to the related loyalty program member.</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>

Field	Details
LoyaltyProgramBadgeId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The loyalty program badge that's associated with the loyalty program member. This field is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgramBadge</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramBadge</p>
LoyaltyProgramMemberId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> The loyalty program member that's associated with the badge. This field is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgramMember</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramMember</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p><b>Description</b> The autogenerated unique identifier of loyalty program member badge.</p>
Reason	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The reason why the member badge was assigned.</p>



Field	Details
StartDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The date from when the loyalty program badge is assigned to the related loyalty program member.</p>
Status	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. Specifies the status of the member badge. Possible values are:</p> <ul style="list-style-type: none"> <li>• Active</li> <li>• Expired</li> </ul>

## 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.

**[LoyaltyProgramMemberBadgeChangeEvent](#) on page 217 (API Version 62.0)**

Change events are available for the object.

**[LoyaltyProgramMemberBadgeHistory](#)**

History is available for tracked fields of the object.

## LoyaltyProgramMemberCase

Represents a junction between a loyalty program member and a case. This object is available in API version 52.0 and later.

## Supported Calls

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

## Fields

Field	Details
CaseId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The Case record associated with the loyalty program member case. This is a relationship field. This field is unique within your organization.</p> <p><b>Relationship Name</b> Case</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Case</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
LoyaltyProgramMemberId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The loyalty program member associated with the loyalty program member case. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgramMember</p> <p><b>Relationship Type</b> Lookup</p>

Field	Details
	<p><b>Refers To</b> LoyaltyProgramMember</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p><b>Description</b> The name of the loyalty program member case.</p>
OwnerId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> The owner of the loyalty program member case. This is a polymorphic relationship field.</p> <p><b>Relationship Name</b> Owner</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Group, User</p>

## 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.

**[LoyaltyProgramMemberCaseChangeEvent](#) on page 217 (API Version 62.0)**

Change events are available for the object.

**[LoyaltyProgramMemberCaseHistory](#)**

History is available for tracked fields of the object.

**[LoyaltyProgramMemberCaseOwnerSharingRule](#)**

Sharing rules are available for the object.

**[LoyaltyProgramMemberCaseShare](#)**

Sharing is available for the object.

## LoyaltyProgramMemberMerge

Represents information about merged loyalty program memberships. This object is available in API version 56.0 and later.

## Supported Calls

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

## Fields

Field	Details
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p><b>Description</b> The autogenerated unique identifier of LoyaltyProgramMemberMerge records.</p>
OwnerId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> ID of the owner of this record. This field is a polymorphic relationship field.</p> <p><b>Relationship Name</b> Owner</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Group, User</p>

Field	Details
Reason	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The reason for merging the memberships.</p>
SourceMemberId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The loyalty program member whose membership is merged with the target loyalty program member.  This field is a relationship field.</p> <p><b>Relationship Name</b> SourceMember</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramMember</p>
SrcMbrDataBody	<p><b>Type</b> base64</p> <p><b>Properties</b> Create, Nillable, Update</p> <p><b>Description</b> The details of the source loyalty program member, such as transaction journals, ledgers, vouchers, tiers, and currencies.</p>
SrcMbrDataContentType	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p><b>Description</b> Specifies the format in which the source member data is stored.  Possible values are:</p> <ul style="list-style-type: none"> <li>• application/html</li> <li>• application/java-archive</li> <li>• application/javascript</li> <li>• application/json</li> </ul>

Field	Details
	<ul style="list-style-type: none"> <li>• application/msword</li> <li>• application/octet-stream</li> <li>• application/octet-stream;type=unknown</li> <li>• application/opx</li> <li>• application/pdf</li> <li>• application/postscript</li> <li>• application/rtf</li> <li>• application/vnd.box.notes</li> <li>• application/vnd.google-apps.document</li> <li>• application/vnd.google-apps.drawing</li> <li>• application/vnd.google-apps.form</li> <li>• application/vnd.google-apps.presentation</li> <li>• application/vnd.google-apps.script</li> <li>• application/vnd.google-apps.spreadsheet</li> <li>• application/vnd.ms-excel</li> <li>• application/vnd.ms-excel.sheet.macroEnabled.12</li> <li>• application/vnd.ms-infopath</li> <li>• application/vnd.ms-powerpoint</li> <li>• application/vnd.ms-powerpoint.presentation.macroEnabled.12</li> <li>• application/vnd.ms-word.document.macroEnabled.12</li> <li>• application/vnd.oasis.opendocument.presentation</li> <li>• application/vnd.oasis.opendocument.spreadsheet</li> <li>• application/vnd.oasis.opendocument.text</li> <li>• application/vnd.openxmlformats-officedocument.presentationml.presentation</li> <li>• application/vnd.openxmlformats-officedocument.presentationml.slideshow</li> <li>• application/vnd.openxmlformats-officedocument.presentationml.template</li> <li>• application/vnd.openxmlformats-officedocument.spreadsheetml.sheet</li> <li>• application/vnd.openxmlformats-officedocument.spreadsheetml.template</li> <li>• application/vnd.openxmlformats-officedocument.wordprocessingml.document</li> <li>• application/vnd.openxmlformats-officedocument.wordprocessingml.template</li> <li>• application/vnd.visio</li> <li>• application/x-gzip</li> <li>• application/x-java-source</li> <li>• application/x-javascript</li> <li>• application/x-shockwave-flash</li> <li>• application/x-sql</li> <li>• application/x-zip-compressed</li> <li>• application/xhtml+xml</li> </ul>

Field	Details
	<ul style="list-style-type: none"><li>• application/xml</li><li>• application/zip</li><li>• audio/mp4</li><li>• audio/mpeg</li><li>• audio/ogg</li><li>• audio/x-aac</li><li>• audio/x-ms-wma</li><li>• audio/x-ms-wmv</li><li>• audio/x-wav</li><li>• image/bmp</li><li>• image/gif</li><li>• image/jpeg</li><li>• image/jpeg;type=mobile</li><li>• image/png</li><li>• image/svg+xml</li><li>• image/tiff</li><li>• image/vnd.adobe.photoshop</li><li>• image/vnd.dwg</li><li>• image/x-photoshop</li><li>• message/rfc822</li><li>• model/vnd.usdz+zip</li><li>• text/css</li><li>• text/csv</li><li>• text/html</li><li>• text/plain</li><li>• text/rtf</li><li>• text/snote</li><li>• text/stypi</li><li>• text/vnd.salesforce.quip-chat</li><li>• text/vnd.salesforce.quip-doc</li><li>• text/vnd.salesforce.quip-sheet</li><li>• text/vnd.salesforce.quip-slides</li><li>• text/vnd.salesforce.quip-template</li><li>• text/vtt</li><li>• text/webviewhtml</li><li>• text/x-c</li><li>• text/x-c++</li><li>• text/xml</li></ul>

Field	Details
	<ul style="list-style-type: none"> <li>• video/mp4</li> <li>• video/mpeg</li> <li>• video/ogg</li> <li>• video/quicktime</li> <li>• video/x-m4v</li> <li>• video/x-ms-asf</li> <li>• video/x-msvideo</li> </ul>
SrcMbrDataLength	<p><b>Type</b> int</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The number of characters in the SrcMbrDataBody field.</p>
SrcMbrDataName	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The unique identifier of the data stored in the SrcMbrDataBody field.</p>
Status	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Restricted picklist, Sort, Update</p> <p><b>Description</b> Required. Specifies the status of merge. Possible values are:</p> <ul style="list-style-type: none"> <li>• Merged</li> <li>• Unmerged</li> </ul>
TargetMemberId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The loyalty program member that the source membership was merged into. This field is a relationship field.</p>



Field	Details
	<b>Relationship Name</b> TargetMember
	<b>Relationship Type</b> Lookup
	<b>Refers To</b> LoyaltyProgramMember

## 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.

### LoyaltyProgramMemberMergeShare

Sharing is available for the object.

## LoyaltyProgramPartner

Represents partners that are associated with a given loyalty program. Partners can be an accrual partner, a redemption partner, or both. This object is available in API version 51.0 and later.

## Supported Calls

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

## Fields

Field	Details
AccrualCostperUnit	<b>Type</b> currency
	<b>Properties</b> Create, Filter, Nillable, Sort, Update
	<b>Description</b> The cost per point associated with the transactions of accrual partner. This field is available in API version 52.0 and later.
AlertValue	<b>Type</b> double
	<b>Properties</b> Create, Filter, Nillable, Sort, Update

Field	Details
	<p><b>Description</b></p> <p>The value that's used to send an alert to a partner to indicate the minimum points required when the billing type is prepaid or the maximum outstanding amount when the billing type is postpaid. This field is available in API version 55.0 and later.</p>
CreditedPoints	<p><b>Type</b></p> <p>double</p> <p><b>Properties</b></p> <p>Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b></p> <p>The number of points credited to partners (using prepaid billing) for member redemptions, accrual cancellations, or any custom processes. This field is available in API version 55.0 and later.</p>
DebitedPoints	<p><b>Type</b></p> <p>double</p> <p><b>Properties</b></p> <p>Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b></p> <p>The total number of points debited from the partner (using prepaid billing) for member accruals, redemption cancellations, or any custom processes. This field is available in API version 55.0 and later.</p>
LastProcessedDateTime	<p><b>Type</b></p> <p>dateTime</p> <p><b>Properties</b></p> <p>Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b></p> <p>The last date and time when the prepaid points balance or the outstanding postpaid bill amount for a partner was calculated. This field is available in API version 55.0 and later.</p>
LastReferencedDate	<p><b>Type</b></p> <p>dateTime</p> <p><b>Properties</b></p> <p>Filter, Nillable, Sort</p> <p><b>Description</b></p> <p>The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b></p> <p>dateTime</p> <p><b>Properties</b></p> <p>Filter, Nillable, Sort</p>

Field	Details
	<p><b>Description</b> The most recent date on which a user viewed this record.</p>
LoyaltyPartnerId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. Account ID of the loyalty program partner. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyPartner</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Account</p>
LoyaltyProgramId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. ID of the loyalty program. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgram</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgram</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Sort, Update</p> <p><b>Description</b> The name of the loyalty program partner.</p>
OwnerId	<p><b>Type</b> reference</p>

Field	Details
	<p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> The ID of the user who owns this record. This is a polymorphic relationship field.</p> <p><b>Relationship Name</b> Owner</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Group, User</p>
PartnerCategory	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The category of the program partnership. Possible values are:</p> <ul style="list-style-type: none"> <li>• Alliance</li> <li>• Bilateral</li> </ul>
PartnerIndustry	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The industry type of the program partner. Picklist values aren't provided for this field and must be added based on the requirements of the organization.</p>
PartnerStatus	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p> <p><b>Description</b> Required. The status of the partner in the loyalty program. The default value is <code>Active</code>. Possible values are:</p> <ul style="list-style-type: none"> <li>• Active</li> <li>• Inactive</li> <li>• Pending</li> </ul>

Field	Details
PartnerType	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Restricted picklist, Sort, Update</p> <p><b>Description</b> Required. The type of the program partnership. Possible values are:</p> <ul style="list-style-type: none"> <li>• Accrual</li> <li>• Both</li> <li>• Redemption</li> </ul>
PartnershipEndDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The date on which the partnership with the program ends.</p>
PartnershipStartDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The date on which the partnership with the program starts.</p>
PointsBalance	<p><b>Type</b> double</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The number of points available with the loyalty program partner using prepaid billing. This field is available in API version 55.0 and later.  This is a calculated field.</p>
BillingType	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Restricted picklist, Sort, Update</p> <p><b>Description</b> Specifies the method in which the loyalty program partner is billed. Possible values are:</p>

Field	Details
	<ul style="list-style-type: none"> <li>• Prepaid</li> <li>• Postpaid</li> </ul> <p>This field is available in API version 55.0 and later.</p>
PostpaidBillAmount	<p><b>Type</b> currency</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The total amount of all credits minus debits for partner transactions from the partnership start date. A positive value indicates that the partner is liable to the program. A negative value indicates the program is liable to the partner. The total amount that either the partner is liable to pay to the program or the program is liable to pay to the partner. This field is available in API version 55.0 and later.</p>
PurchasedPoints	<p><b>Type</b> double</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The total number of points that the partner (using prepaid billing) has purchased across all their packs. This field is available in API version 55.0 and later.</p>
RedemptionCostperUnit	<p><b>Type</b> currency</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The cost per point associated with the transactions of redemption partner. This field is available in API version 52.0 and later.</p>

## 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.

**[LoyaltyProgramPartnerChangeEvent](#) on page 217 (API Version 62.0)**

Change events are available for the object.

**[LoyaltyProgramPartnerHistory](#) (API version 52.0)**

History is available for tracked fields of the object.

**[LoyaltyProgramPartnerOwnerSharingRule](#)**

Sharing rules are available for the object.

**LoyaltyProgramPartnerShare (API version 52.0)**

Sharing is available for the object.

## LoyaltyProgramPartnerLedger

---

Represents the ledger used by a loyalty program to credit points to or debit points from a prepaid partner's points pack or from a postpaid partner's point balance. This ledger can also be used to add or deduct the monetary value of points from the partner's postpaid bill. This object is available in API version 55.0 and later.

### Supported Calls

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

### Fields

Field	Details
ActivityDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Create, Filter, Sort, Update</p> <p><b>Description</b> Required. The date on which the loyalty program member transacts with the loyalty program partner.</p>
LoyaltyLedgerId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. Specifies the type of the loyalty program partner ledger.  This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyLedger</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyLedger</p>
LoyaltyProgramMemberId	<p><b>Type</b> reference</p>

Field	Details
	<p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The loyalty program member that's related to the loyalty program partner ledger. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgramMember</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramMember</p>
LoyaltyProgramPartnerId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The loyalty program partner that's related to the loyalty program partner ledger. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgramPartner</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramPartner</p>
Points	<p><b>Type</b> double</p> <p><b>Properties</b> Create, Filter, Sort, Update</p> <p><b>Description</b> Required. The number of points credited to or debited from the loyalty program partner's points balance or postpaid account balance.</p>
PromotionId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The promotion that's related to the loyalty program partner ledger. This is a relationship field.</p>



Field	Details
	<p><b>Relationship Name</b> Promotion</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Promotion</p>
TransactionJournalId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The transaction journal that's related to the loyalty program partner ledger. This is a relationship field.</p> <p><b>Relationship Name</b> TransactionJournal</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> TransactionJournal</p>
Type	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p> <p><b>Description</b> Required. Specifies the type of the loyalty program partner ledger. Possible values are:</p> <ul style="list-style-type: none"> <li>• Credit</li> <li>• CreditCancellation—Credit Cancellation</li> <li>• Debit</li> <li>• DebitCancellation—Debit Cancellation</li> </ul> <p>The default value is Credit.</p>

## 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.

### [LoyaltyProgramPartnerLedgerChangeEvent](#)

Change events are available for the object.

# LoyaltyProgramProcess

---

Represents information about loyalty program processes that determine how transaction journals are processed. This object is available in API version 54.0 and later.

## Supported Calls

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

## Fields

Field	Details
Description	<p><b>Type</b> textarea</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The description of the loyalty program process.</p>
ExecutionType	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Restricted picklist, Sort</p> <p><b>Description</b> Specifies the method used for executing the loyalty program process. Determines how transaction journals are processed. If you select Real Time, incoming transactions are processed as soon as they're created. If you select Batch, then Run Program Process for Transaction Journal action in Flows is used to schedule when pending transaction journals are processed.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>• Batch</li> <li>• BatchAndRealTime—Both</li> <li>• RealTime</li> </ul> <p>The default value is RealTime.</p>
JournalSubTypeId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> The ID of the journal subtype record associated with the loyalty program process record. This is a relationship field.</p>

Field	Details
	<p><b>Relationship Name</b> JournalSubType</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> JournalSubType</p>
JournalTypeId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> The ID of the journal type record associated with the loyalty program process record. This is a relationship field.</p> <p><b>Relationship Name</b> JournalType</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> JournalType</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
LoyaltyProgramId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> The loyalty program associated with the loyalty program process record.</p>

Field	Details
	<p>This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgram</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgram</p>
LoyaltyTierGroup	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> The tier group of a program that's associated with the loyalty program process. This field is available in API version 56.0 and later. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgram</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgram</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Sort, Update</p> <p><b>Description</b> Required. The name of the loyalty program process.</p>
OwnerId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> The ID of the user who owns this record. This is a polymorphic relationship field.</p> <p><b>Relationship Name</b> Owner</p> <p><b>Relationship Type</b> Lookup</p>

Field	Details
	<p><b>Refers To</b> Group, User</p>
ProcessType	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Restricted picklist, Sort</p> <p><b>Description</b> Required. Specifies the type of loyalty program process. Possible values are:</p> <ul style="list-style-type: none"> <li>• <code>CancelVoucher</code>—This value is available in API version 56.0 and later.</li> <li>• <code>CreditPointsToMembers</code>—This value is available in API version 56.0 and later.</li> <li>• <code>DebitPointsFromMembers</code>—This value is available in API version 56.0 and later.</li> <li>• <code>EnrollInPromotion</code>—This value is available in API version 56.0 and later.</li> <li>• <code>GetMemberPromotions</code>—This value is available in API version 56.0 and later.</li> <li>• <code>IssueVoucher</code>—This value is available in API version 56.0 and later.</li> <li>• <code>TierProcessing</code>—This value is available in API version 56.0 and later.</li> <li>• <code>TransactionJournal</code></li> <li>• <code>UnenrollMember</code>—This value is available in API version 56.0 and later.</li> <li>• <code>UpdateMemberDetails</code>—This value is available in API version 56.0 and later.</li> <li>• <code>OptOutPromotion</code>—This value is available in API version 58.0 and later.</li> </ul>
Status	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p> <p><b>Description</b> Specifies the status of the loyalty program process. Possible values are:</p> <ul style="list-style-type: none"> <li>• <code>Active</code></li> <li>• <code>Draft</code></li> <li>• <code>Inactive</code></li> </ul> <p>The default value is <code>Draft</code>.</p>

## 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.

**LoyaltyProgramProcessHistory**

History is available for tracked fields of the object.

**LoyaltyProgramProcessOwnerSharingRule**

Sharing rules are available for the object.

**LoyaltyProgramProcessShare**

Sharing is available for the object.

## LoyaltyProgramWidget

---

Represents the configuration of a widget that a user can apply to the out-of-the-box widget templates and FlexCards provided by Omnistudio. The configuration helps implement custom UI changes, such as promotion type, promotion tag line, and so on for a user. This object is available in API version 57.0 and later.

### Supported Calls

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

### Fields

Field	Details
CustomizedContent	<p><b>Type</b> textarea</p> <p><b>Properties</b> Create, Nillable, Update</p> <p><b>Description</b> The detail about the customization that's applied to a widget. For example, promotion_title: Great Times</p>
Description	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The description of the widget.</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>

Field	Details
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p><b>Description</b> The name of the widget.</p>
OwnerId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> The owner of the widget. This field is a polymorphic relationship field.</p> <p><b>Relationship Name</b> Owner</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Group, User</p>
ProgramId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> Required. The identifier of the loyalty program that's associated with the widget. This field is a relationship field.</p> <p><b>Relationship Name</b> Program</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgram</p>

Field	Details
TemplateName	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> Required. The name of the widget template.</p>
TemplateVersion	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> Required. The version of the widget template.</p>

## 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.

### LoyaltyProgramWidgetShare

Sharing is available for the object.

## LoyaltyTier

Represents information about tiers of a loyalty program. A tier group has a set of tiers in a sequential order. This object is available in API version 51.0 and later.

You can set up Loyalty Tiers to group the members based on their qualifying points. Qualifying points are awarded to members based on their activities in a loyalty program. The hierarchies are created in the tiers using the sequence number. The qualifying points are used to assess a member's tier for upgrade or downgrade.

## Supported Calls

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

## Fields

Field	Details
Color	<p><b>Type</b> string</p>



Field	Details
	<p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The color that represents the tier. This field is available in API version 62.0 and later.</p>
Description	<p><b>Type</b> textarea</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> Detailed description of the tier.</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
LoyaltyTierGroupId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> ID of the loyalty tier group. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyTierGroup</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyTierGroup</p>

Field	Details
MaximumEligibleBalance	<p><b>Type</b> double</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The maximum balance that's required for loyalty program members to be eligible for the tier. This field is available in API version 58.0 and later.</p>
MinimumEligibleBalance	<p><b>Type</b> double</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The minimum balance that's required for loyalty program members to be eligible for the tier. This field is available in API version 58.0 and later.</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Sort, Update</p> <p><b>Description</b> The name of the loyalty tier.</p>
SequenceNumber	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Nillable, Sort, Update</p> <p><b>Description</b> The sequence number indicates the ranking of the tier. The tier with the highest sequence number indicates the highest tier.</p>

## 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.

### [LoyaltyTierChangeEvent](#) on page 217 (API Version 62.0)

Change events are available for the object.

### [LoyaltyTierHistory](#)

History is available for tracked fields of the object.

## LoyaltyTierBenefit

---

A junction object that associates the loyalty benefits with the loyalty program tiers. This object is available in API version 51.0 and later.

### Supported Calls

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

### Fields

Field	Details
BenefitId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. Benefit associated with the loyalty tier benefit. This is a relationship field.</p> <p><b>Relationship Name</b> Benefit</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Benefit</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
LoyaltyTierId	<p><b>Type</b> reference</p>

Field	Details
	<p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> Tier associated with the loyalty benefit. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyTier</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyTier</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Sort, Update</p> <p><b>Description</b> The name of the loyalty tier benefit.</p>

## 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.

**[LoyaltyTierBenefitChangeEvent](#) on page 217 (API Version 62.0)**

Change events are available for the object.

## LoyaltyTierGroup

Represents information about tier groups of a loyalty program. Tier group categorizes members based on their loyalty points. This object is available in API version 51.0 and later.

## Supported Calls

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

## Fields

Field	Details
Description	<p><b>Type</b> textarea</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> Detailed description of the tier group.</p>
ExtendExpiration	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p> <p><b>Description</b> Required. The field decides if the tier is changed on the exact day or the last day of the month. Possible values are:</p> <ul style="list-style-type: none"> <li>• <code>MemberEnrollmentAnniversary</code>— Select this option only when TierModel is Anniversary. This change is available from API version 52.0 and later.</li> <li>• <code>None</code></li> <li>• <code>QualifyingPointsResetDate</code>— This value can only be selected when TierModel is Fixed. This change is available from API version 52.0 and later.</li> <li>• <code>TillEndOfMonth</code>— This value isn't supported in API version 52.0 and later.</li> </ul>
IsActive	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. Indicates whether the loyalty tier group is active (<code>true</code>) or not (<code>false</code>).</p>
IsHiddenToMembers	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> Indicates whether the loyalty tier group is hidden (<code>true</code>) or not (<code>false</code>).</p>
IsPrimary	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p>

Field	Details
	<p><b>Description</b> Indicates whether the loyalty tier group is primary (<code>true</code>) or not (<code>false</code>).</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
LoyaltyProgramId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> ID of the loyalty program. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgram</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgram</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Sort, Update</p> <p><b>Description</b> The name of the loyalty tier group.</p>
QpResetDate	<p><b>Type</b> date</p>

Field	Details
	<p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> Required. Start date of the QP reset process.</p>
QpResetFrequency	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The frequency at which members' qualifying points must be reset.</p>
QpResetPeriod	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Restricted picklist, Sort, Update</p> <p><b>Description</b> Required. Unit of the period for QP reset. Possible values are:</p> <ul style="list-style-type: none"> <li>• <code>Days</code>—This value isn't supported in API version 52.0 and later.</li> <li>• <code>Months</code>— This value isn't supported in API version 52.0 and later.</li> <li>• <code>Years</code></li> </ul>
TierAssessmentCurrencyId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The ID of the Loyalty Program Currency that's used to assess and assign tiers to members of the loyalty program. Members are assigned tiers based on their points balance for the currency. This field is available in API version 58.0 and later. This field is a relationship field.</p> <p><b>Relationship Name</b> TierAssessmentCurrency</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramCurrency</p>
TierModel	<p><b>Type</b> picklist</p>

Field	Details
	<p><b>Properties</b> Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p> <p><b>Description</b> Required. It's the model that decides the start date of the qualifying period.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>• Anniversary</li> <li>• Fixed</li> </ul>
TierPeriodQuantity	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The period after which members' tier must be reset.</p>
TierPeriodUnit	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Restricted picklist, Sort, Update</p> <p><b>Description</b> Required. The frequency type for the tier assessment.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>• Days—This value isn't supported in API version 52.0 and later.</li> <li>• Months</li> <li>• Years</li> </ul>

## 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.

**[LoyaltyTierGroupChangeEvent](#) on page 217 (API Version 62.0)**

Change events are available for the object.

## MemberBenefit

Represents details about a benefit that's assigned to a loyalty program member and has a benefit action that's triggered. This object is available in API version 51.0 and later.



## Supported Calls

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

## Fields

Field	Details
AccountId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The ID of the account associated with the member benefit. This is a relationship field.</p> <p><b>Relationship Name</b> Account</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Account</p>
BenefitId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The ID of the benefit associated with the member benefit. This is a relationship field.</p> <p><b>Relationship Name</b> Benefit</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Benefit</p>
ContactId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The ID of the contact associated with the member benefit.</p>

Field	Details
	<p>This is a relationship field.</p> <p><b>Relationship Name</b> Contact</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Contact</p>
Description	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The description of the member benefit.</p>
EndDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The end date of the member benefit.</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
MemberId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p>

Field	Details
	<p><b>Description</b> The ID of the member associated with the member benefit. This is a polymorphic relationship field.</p> <p><b>Relationship Name</b> Member</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramMember</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Sort, Update</p> <p><b>Description</b> The name of the member benefit.</p>
PrimaryRefObjectRecord	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The ID of a record created by the member benefit.</p>
PrimaryRefObjectRecordTitle	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The title of a record created by the member benefit.</p>
SourceType	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> Required. The source of the benefit. Possible values are:</p> <ul style="list-style-type: none"> <li>• LoyaltyTierProcessing</li> <li>• Manual</li> </ul>

Field	Details
StartDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The start date of the member benefit.</p>
Status	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p> <p><b>Description</b> Required. The status of the member benefit.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>Expired</li> <li>ExpiredPendingProcessing—Expired - Awaiting Processing</li> <li>NewPendingProcessing—New - Awaiting Processing</li> <li>Processed</li> <li>UpdatedPendingProcessing—Updated - Awaiting Processing</li> </ul> <p>The default value is <code>NewPendingProcessing</code>.</p>

## 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.

### [MemberBenefitChangeEvent](#) on page 217 (API Version 62.0)

Change events are available for the object.

### [MemberBenefitHistory](#)

History is available for tracked fields of the object.

## NetworkReferencedObject

A junction object that associates the loyalty program with the Experience Cloud site. This object is available in API version 51.0 and later.

## Supported Calls

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

## Fields

Field	Details
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p><b>Description</b> Unique identifier for the Network Referenced Object.</p>
NetworkId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The Id of the Experience Cloud site associated with the loyalty program. This is a relationship field.</p> <p><b>Relationship Name</b> Network</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Network</p>
ReferencedObjectId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p>

Field	Details
	<p><b>Description</b> The object associated with the Experience Cloud site. This is a polymorphic relationship field.</p> <p><b>Relationship Name</b> ReferencedObject</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgram</p>

## Promotion

Represents a promotion associated with the loyalty program. It captures the information such as type of the promotion, issue of a voucher or credit points, promotion enrollment, and status of the promotion. This object is available in API version 51.0 and later.

## Supported Calls

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

## Fields

Field	Details
BusinessUnitId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> ID of the internal organization unit that contains the details of the Marketing Cloud business unit associated with the promotion. This field is a relationship field.</p> <p><b>Relationship Name</b> BusinessUnit</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> InternalOrganizationUnit</p>

Field	Details
CampaignId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The campaign associated with the promotion. This is a relationship field.</p> <p><b>Relationship Name</b> Campaign</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Campaign</p>
CumulativeUsageTarget	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The number of times a member must participate or the amount of money spent to qualify for a reward.</p>
CurrentLiabilityAmount	<p><b>Type</b> currency</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The liability amount incurred for the promotion until now. This field is available in API version 52.0 and later.</p>
CurrentPointCount	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The liability amount incurred for the promotion until now. This field is available in API version 52.0 and later.</p>
DefaultVoucherDefinitionId	<p><b>Type</b> reference</p>

Field	Details
	<p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The voucher associated with the promotion. This is a relationship field.</p> <p><b>Relationship Name</b> DefaultVoucherDefinition</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> VoucherDefinition</p>
Description	<p><b>Type</b> textarea</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> Description of the promotion.</p>
DisplayName	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The display name of the promotion. This field is available in API version 60.0 and later where Global Promotions Management is enabled.</p>
EndDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> End date of the promotion.</p>
EndTime	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The end date and time of the promotion. This field is available in API version 60.0 and later where Unified Promotions is enabled.</p>



Field	Details
EnrollmentEndDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The last date to enroll in the promotion.</p>
EnrollmentStartDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The first date to enroll in the promotion.</p>
FulfillmentAction	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p><b>Description</b> Select how a loyalty member is rewarded for fulfilling promotion requirements. Possible values are:</p> <ul style="list-style-type: none"> <li>• <code>CreditPoints</code>—Credit Points</li> <li>• <code>IssueVoucher</code>—Issue Voucher</li> </ul>
GameDefinitionId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> ID of the game definition associated with the promotion. This field is a relationship field.</p> <p><b>Relationship Name</b> GameDefinition</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> GameDefinition</p>
ImageUrl	<p><b>Type</b> url</p>

Field	Details
	<p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The URL for the promotion image. This field is available in API version 57.0 and later.</p>
IsActive	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> Indicates whether the promotion is active. This field is available in API version 52.0 and later.</p>
IsCloned	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> Indicates whether the promotion has been created by cloning an existing promotion. This field is available in API version 61.0 and later.  The default value is <code>false</code>.</p>
IsEnrollmentRequired	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> Indicates whether enrollment is required to participate in the promotion.</p>
IsPointsWithFactor	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> Indicates whether the promotion is applied by multiplying points with a factor or by using the fixed value in the TotalRewardPoints field. This field is available in API version 52.0 and later.  The default value is <code>False</code>.</p>
IsQuickPromotion	<p><b>Type</b> boolean</p>

Field	Details
	<p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> Indicates whether the records represent a quick promotion (<code>true</code>) or not (<code>false</code>). This field is available in API version 57.0 and later.  The default value is <code>false</code>.</p>
IssuedVoucherCount	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The number of vouchers issued for the promotion until current date. This field is available in API version 52.0 and later.</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
LoyaltyProgramCurrencyId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The loyalty program currency associated with the promotion.  This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgramCurrency</p> <p><b>Relationship Type</b> Lookup</p>

Field	Details
	<p><b>Refers To</b> LoyaltyProgramCurrency</p>
LoyaltyProgramId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The loyalty program associated with the promotion. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgram</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgram</p>
LoyaltyPromotionType	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p><b>Description</b> The type of promotion. Possible values are:</p> <ul style="list-style-type: none"> <li>• Cumulative</li> <li>• Standard</li> <li>• Joint</li> </ul> <p>This field is available in API version 55.0 and later.</p>
MaximumLiabilityAmount	<p><b>Type</b> currency</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The maximum liability amount that can be incurred for the promotion.</p>
MaximumRewardValue	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p>

Field	Details
	<p><b>Description</b> The total value that can be accumulated for the promotion.</p>
MaximumVoucherCount	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The maximum number of vouchers that can be issued for the promotion. This field is available in API version 52.0 and later.</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Sort, Update</p> <p><b>Description</b> Unique identifier for the promotion.</p>
Objective	<p><b>Type</b> textarea</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> Business objective of the promotion. For example, launching a new product, keeping customers satisfied, or clearing stock.</p>
OwnerId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> The ID of the user who owns this record. This is a polymorphic relationship field.</p> <p><b>Relationship Name</b> Owner</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Group, User</p>

Field	Details
PointFactor	<p><b>Type</b> double</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The factor with which points are multiplied or divided when <code>IsPointsWithFactor</code> is set to true. This field is available in API version 52.0 and later.</p>
PriorityNumber	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The priority number of the promotion. This field is available in API version 60.0 and later, where Global Promotions Management is enabled.</p>
ProductCatalogId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> ID of the product catalog associated with the promotion. This field is available in API version 60.0 and later, where Global Promotions Management is enabled.  This field is a relationship field.</p> <p><b>Relationship Name</b> ProductCatalog</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> ProductCatalog</p>
PromotionCode	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Nillable, Sort, Update</p> <p><b>Description</b> The unique code of the promotion. This field is available in API version 60.0 and later, where Global Promotions Management is enabled.</p>
PromotionConfiguration	<p><b>Type</b> textarea</p>

Field	Details
	<p><b>Properties</b> Create, Nillable, Update</p> <p><b>Description</b> The configuration of a quick promotion. This field is available in API version 57.0 and later.</p>
RelatedPromotionId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> ID of the promotion that was cloned to create the promotion. This field is a relationship field.</p> <p><b>Relationship Name</b> RelatedPromotion</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Promotion</p>
RuleLibraryId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> ID of the promotion that was cloned to create the promotion. This field is available in API version 62.0 and later. This field is a relationship field.</p> <p><b>Relationship Name</b> RelatedPromotion</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Promotion</p>
StartDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The start date of the promotion.</p>

Field	Details
StartDateTime	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The start date and time of the promotion. This field is available in API version 60.0 and later where Global Promotions Management is enabled.</p>
Status	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p><b>Description</b> The status of the promotion. Possible values are:</p> <ul style="list-style-type: none"><li>• Cancelled</li><li>• Complete</li><li>• Draft</li><li>• InProgress</li></ul>
TotalRewardPoints	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The points awarded on availing the promotion.</p>
TransactionJournalType	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p><b>Description</b> The journal type that has been applied for the promotion. This field is available in API version 52.0 and later. Possible values are:</p> <ul style="list-style-type: none"><li>• Accrual</li><li>• Redemption</li></ul>



## 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.

### PromotionChangeEvent

Change events are available for the object.

### PromotionFeed

Feed tracking is available for the object.

### PromotionHistory

History is available for tracked fields of the object.

### PromotionOwnerSharingRule

Sharing rules are available for the object.

### PromotionShare

Sharing is available for the object.

## PromotionActionableList

---

Represent of a junction between a promotion and an actionable list. 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

The PromotionActionableList object is available in orgs that have the Actionable Segmentation setting enabled.

## Fields

Field	Details
ActionableListId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The ID of the actionable list associated with the promotion actionable list. This field is a relationship field.</p> <p><b>Relationship Name</b> ActionableList</p> <p><b>Refers To</b> ActionableList</p>

Field	Details
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p><b>Description</b> The autogenerated number of the promotion actionable list.</p>
PromotionId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> The ID of the promotion associated with the promotion actionable list. This field is a relationship field.</p> <p><b>Relationship Name</b> Promotion</p> <p><b>Relationship Type</b> Master-detail</p> <p><b>Refers To</b> Promotion (the master object)</p>

## PromotionChannel

Represents a promotion's eligible channel. 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

PromotionLimit is only available in orgs where Global Promotions Management is enabled.

## Fields

Field	Details
EndDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The date from which the channel is eligible for the promotion.</p>
EngagementChannelTypeId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> ID of the engagement channel type that the channel belongs to. This field is a relationship field.</p> <p><b>Relationship Name</b> EngagementChannelType</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> EngagementChannelType</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p>

Field	Details
	<p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p><b>Description</b> The autogenerated unique number of the record.</p>
PromotionId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> ID of the promotion for which the channel is eligible. This field is a relationship field.</p> <p><b>Relationship Name</b> Promotion</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Promotion</p>
RetailLocationGroupId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> ID of the retail location group that's eligible for the promotion. This field is a relationship field.</p> <p><b>Relationship Name</b> RetailLocationGroup</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> RetailLocationGroup</p>

Field	Details
RetailStoreId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> ID of the retail store that's eligible for the promotion. This field is a relationship field.</p> <p><b>Relationship Name</b> RetailStore</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> RetailStore</p>
StartDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The date till which the channel is eligible for the promotion.</p>

## 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.

### PromotionChannelFeed

Feed tracking is available for the object.

### PromotionChannelHistory

History is available for tracked fields of the object.

## PromotionLimit

Represents the limit of a promotion. 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

PromotionLimit is only available in orgs where Global Promotions Management is enabled.

## Fields

Field	Details
Category	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Restricted picklist, Sort, Update</p> <p><b>Description</b> Required. The category of promotion limit. Possible values are:</p> <ul style="list-style-type: none"> <li>• CustomerLevel</li> <li>• PromotionLevel</li> </ul>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
LoyaltyProgramCurrencyId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> ID of the loyalty program currency for which the promotion limit is set. Specify a loyalty program currency when the limit type is PointsCredited. This field is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgramCurrency</p>

Field	Details
	<p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramCurrency</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p><b>Description</b> The autogenerated unique number of the record.</p>
PromotionId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> ID of the promotion that the limit is for. This field is a relationship field.</p> <p><b>Relationship Name</b> Promotion</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Promotion</p>
Type	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Restricted picklist, Sort, Update</p> <p><b>Description</b> Required. The type of promotion limit. Possible values are:</p> <ul style="list-style-type: none"> <li>• CartItemCount</li> <li>• FinancialLiability</li> <li>• OrderCount</li> <li>• PointsCredited</li> <li>• UsagePerCustomer</li> <li>• VouchersIssued</li> </ul>

Field	Details
Value	<p><b>Type</b> double</p> <p><b>Properties</b> Create, Filter, Sort, Update</p> <p><b>Description</b> Required. The value of the promotion limit.</p>

## 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.

### PromotionLimitFeed

Feed tracking is available for the object.

### PromotionLimitHistory

History is available for tracked fields of the object.

## PromotionLimitUsage

Represents information about the total usage of a promotion limit. 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

PromotionLimitUsage is only available in orgs where Global Promotions Management is enabled.

## Fields

Field	Details
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nullable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>



Field	Details
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p><b>Description</b> The autogenerated unique number of the record.</p>
PromotionLimitId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> ID of the promotion limit for which the usage is tracked. This field is a relationship field.</p> <p><b>Relationship Name</b> PromotionLimit</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> PromotionLimit</p>
Usage	<p><b>Type</b> double</p> <p><b>Properties</b> Create, Filter, Sort, Update</p> <p><b>Description</b> Required. The usage of the promotion limit till date.</p>

## 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.

### PromotionLimitUsageFeed

Feed tracking is available for the object.

**PromotionLimitUsageHistory**

History is available for tracked fields of the object.

## PromotionLoyaltyPtnrProd

---

Represents a relationship between a promotion and a loyalty partner product. It indicates the products or product categories that are applied to a particular promotion. This object is available in API version 51.0 and later.

### Supported Calls

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

### Fields

Field	Details
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
LoyaltyPartnerId	<p><b>Type</b> reference</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The loyalty partner associated with the promotion. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyPartner</p> <p><b>Relationship Type</b> Lookup</p>

Field	Details
	<p><b>Refers To</b> LoyaltyProgramPartner</p>
LoyaltyPartnerProductId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The loyalty partner product associated with the promotion loyalty partner product. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyPartnerProduct</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyPartnerProduct</p>
LoyaltyProductId	<p><b>Type</b> reference</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The loyalty product associated with the promotion. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProduct</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Product2</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p><b>Description</b> Unique identifier for the promotion loyalty partner product.</p>
ProductCategoryId	<p><b>Type</b> reference</p>

Field	Details
	<p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The product category associated with the promotion. This is a relationship field.</p> <p><b>Relationship Name</b> ProductCategory</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> ProductCategory</p>
PromotionId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> The promotion associated with the promotion loyalty partner product. This is a relationship field.</p> <p><b>Relationship Name</b> Promotion</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Promotion</p>

## 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.

**[PromotionLoyaltyPtnrProdChangeEvent](#) on page 217 (API Version 62.0)**

Change events are available for the object.

**[PromotionLoyaltyPtnrProdFeed](#)**

Feed tracking is available for the object.

**[PromotionLoyaltyPtnrProdHistory](#)**

History is available for tracked fields of the object.

## PromotionMarketSegment

---

Represents a relationship between a promotion and Data Cloud segment. It indicates the segments targeted for a particular promotion. This object is available in API version 51.0 and later.

### Supported Calls

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

### Fields

Field	Details
CurrencyIsoCode	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p><b>Description</b> The currency ISO code of the promotion. This field is only available in orgs where multiple currencies are enabled.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>• USD—U.S. Dollar</li> </ul> <p>The default value is USD.</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
MarketSegmentId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p>

Field	Details
	<p><b>Description</b> The ID of the market segment associated with the promotion market segment.</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p><b>Description</b> Unique identifier for the promotion market segment.</p>
PromotionId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> The ID of the promotion associated with the promotion market segment. This is a relationship field.</p> <p><b>Relationship Name</b> Promotion</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Promotion</p>
PromotionSegmentId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The ID of the related promotion segment. This field is accessible in orgs with a Commerce Cloud license, where users have access to Promotions. The field associates a promotion with its segments. Promotion segments are different from Data Cloud segments. This field is a relationship field.</p> <p><b>Relationship Name</b> PromotionSegment</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> PromotionSegment</p>

## 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.

### [PromotionMarketSegmentChangeEvent](#) on page 217 (API Version 62.0)

Change events are available for the object.

### [PromotionMarketSegmentFeed](#)

Feed tracking is available for the object.

### [PromotionMarketSegmentHistory](#)

History is available for tracked fields of the object.

## PromotionPartyUsage

---

Represents the details of number of times a promotion has been used by a customer. 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

PromotionPartyUsage is only available in orgs where Global Promotions Management is enabled.

## Fields

Field	Details
AccountId	<p><b>Type</b> reference</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> ID of the account associated with the organization that has used the promotion. This field is a relationship field.</p> <p><b>Relationship Name</b> Account</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Account</p>

Field	Details
ContactId	<p><b>Type</b> reference</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> ID of the contact associated with the person who has used the promotion. This field is a relationship field.</p> <p><b>Relationship Name</b> Contact</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Contact</p>
IssuedVoucherCount	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The number of vouchers that have been issued to the customer as the associated promotion's reward.</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
LoyaltyProgramMemberId	<p><b>Type</b> reference</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p>



Field	Details
	<p><b>Description</b> ID of the loyalty program member who has used the promotion. This field is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgramMember</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramMember</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p><b>Description</b> The autogenerated unique number of the record.</p>
OrderCount	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The number of orders for which the customer has applied the promotion.</p>
PartyId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> ID of the person or the organization that has been provided the discount. This field is a polymorphic relationship field.</p> <p><b>Relationship Name</b> Party</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Account, Contact, LoyaltyProgramMember</p>
PointsCredited	<p><b>Type</b> double</p>

Field	Details
	<p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The number of points that have been credited to a loyalty program member as the associated promotion's reward.</p>
PromotionId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. ID of the promotion for which the customer's usage is tracked. This field is a relationship field.</p> <p><b>Relationship Name</b> Promotion</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Promotion</p>
Type	<p><b>Type</b> picklist</p> <p><b>Properties</b> Filter, Group, Nillable, Restricted picklist, Sort</p> <p><b>Description</b> The type of customer for whom the promotion usage is tracked. Possible values are:</p> <ul style="list-style-type: none"> <li>• Account</li> <li>• Contact</li> <li>• LoyaltyProgramMember</li> </ul>

## 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.

### PromotionPartyUsageChangeEvent

Change events are available for the object.

### PromotionPartyUsageFeed

Feed tracking is available for the object.

**PromotionPartyUsageHistory**

History is available for tracked fields of the object.

## PromotionPartyUsageDetail

---

Represents the amount that's discounted by a promotion for a transaction journal. 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

PromotionPartyUsageDetail is only available in orgs where Global Promotions Management is enabled.

### Fields

Field	Details
AccountId	<p><b>Type</b> reference</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> ID of the account associated with the organization that has used the promotion. This field is a relationship field.</p> <p><b>Relationship Name</b> Account</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Account</p>
ContactId	<p><b>Type</b> reference</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> ID of the contact associated with the person who has used the promotion. This field is a relationship field.</p> <p><b>Relationship Name</b> Contact</p>

Field	Details
	<p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Contact</p>
DiscountAmount	<p><b>Type</b> currency</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The monetary discount provided to the person or organization who has availed the promotion.</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
LoyaltyProgramMemberId	<p><b>Type</b> reference</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> ID of the loyalty program member who has used the promotion. This field is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgramMember</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramMember</p>

Field	Details
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p><b>Description</b> The autogenerated unique number of the record.</p>
PartyId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> ID of the person or the organization that has been provided the discount. This field is a polymorphic relationship field.</p> <p><b>Relationship Name</b> Party</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Account, Contact, LoyaltyProgramMember</p>
PartyType	<p><b>Type</b> picklist</p> <p><b>Properties</b> Filter, Group, Nillable, Restricted picklist, Sort</p> <p><b>Description</b> The type of customer that has been provided the discount. Possible values are:</p> <ul style="list-style-type: none"> <li>• Account</li> <li>• Contact</li> <li>• LoyaltyProgramMember</li> </ul>
PromotionId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The promotion that has provided the discount to the customer. This field is a relationship field.</p>

Field	Details
	<p><b>Relationship Name</b> Promotion</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Promotion</p>
TransactionJournalId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> ID of the transaction journal that represents the customer activity for which the promotion has been applied.  This field is a relationship field.</p> <p><b>Relationship Name</b> TransactionJournal</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> TransactionJournal</p>

## PromotionProduct

---

Represents a junction between a promotion and an product that's eligible for the promotion. 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

PromotionProduct is available in orgs where Global Promotions Management is enabled.

## Fields

Field	Details
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p><b>Description</b> The autogenerated unique number of the record.</p>
ProductId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. ID of the product that's eligible for the associated promotion. This field is a relationship field.</p> <p><b>Relationship Name</b> Product</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Product2</p>
PromotionId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p>

Field	Details
	<p><b>Description</b> The promotion for which eligible product is selected. This field is a relationship field.</p> <p><b>Relationship Name</b> Promotion</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Promotion</p>

## 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.

### PromotionProductFeed

Feed tracking is available for the object.

### PromotionProductHistory

History is available for tracked fields of the object.

## PromotionProductCategory

Represents a junction between a promotion and a product category that's eligible for the promotion. 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

PromotionProductCategory is only available in orgs where Global Promotions Management is enabled.

## Fields

Field	Details
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p>



Field	Details
	<p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p><b>Description</b> The autogenerated unique number of the record.</p>
ProductCategoryId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. of the product category that's eligible for the associated promotion. This field is a relationship field.</p> <p><b>Relationship Name</b> ProductCategory</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> ProductCategory</p>
PromotionId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> The promotion for which eligible product category is selected. This field is a relationship field.</p> <p><b>Relationship Name</b> Promotion</p>

Field	Details
	<b>Relationship Type</b> Lookup
	<b>Refers To</b> Promotion

## 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.

### PromotionProductCategoryFeed

Feed tracking is available for the object.

### PromotionProductCategoryHistory

History is available for tracked fields of the object.

## PromotionRuleTemplate

Represents the configuration of a promotion rule template. This object is available in API version 60.0 and later.

## Supported Calls

`describeSObjects()`, `query()`

## Special Access Rules

PromotionRuleTemplate is only available in orgs where Global Promotions Management is enabled. The object isn't available in the Object Manager and can't be customized.

## Fields

Field	Details
ApiName	<b>Type</b> string
	<b>Properties</b> Filter, Group, Nillable, Sort
	<b>Description</b> The API name of the promotion rule template.
Category	<b>Type</b> string

Field	Details
	<p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The category of the promotion rule template.</p>
Description	<p><b>Type</b> textarea</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The description of the promotion rule template.</p>
ExternalId	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> Reserved for internal use.</p>
IconUrl	<p><b>Type</b> url</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The URL of the promotion rule template's icon.</p>
JournalTypeId	<p><b>Type</b> reference</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The ID of the journal type that represents the type of customer activity the promotion rule template is applicable for.  This field is a relationship field.</p> <p><b>Relationship Name</b> JournalType</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> JournalType</p>

Field	Details
Label	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The name of the promotion rule template.</p>
RuleConfiguration	<p><b>Type</b> textarea</p> <p><b>Properties</b> Nillable</p> <p><b>Description</b> The promotion rule configuration in JSON format.</p>
RuleRewardConfiguration	<p><b>Type</b> textarea</p> <p><b>Properties</b> Nillable</p> <p><b>Description</b> The rule reward configuration details in JSON format, which are applicable for the promotion rule template.</p>

## PromotionStage

Represents a stage when the promotion sends an email to the promotion's target audience. With Global Promotions Management, a promotion launch email is sent after the promotion is activated. 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

PromotionStage is only available in orgs where Global Promotions Management is enabled.

## Fields

Field	Details
ApiName	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Sort, Update</p> <p><b>Description</b> Required. The unique name of the promotion stage.</p>
Configuration	<p><b>Type</b> textarea</p> <p><b>Properties</b> Create, Nillable, Update</p> <p><b>Description</b> The configuration of the promotion stage.</p>
Description	<p><b>Type</b> textarea</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The description of the promotion stage.</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Sort, Update</p>

Field	Details
	<p><b>Description</b> The name of the promotion stage.</p>
PromotionId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> The ID of the promotion that the stage belongs to. This field is a relationship field.</p> <p><b>Relationship Name</b> Promotion</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Promotion</p>
SequenceNumber	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The sequence number of the stage in a promotion. The sequence number of the Promotion Launched stage is 1.</p>
Type	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Restricted picklist, Sort, Update</p> <p><b>Description</b> Required. The type of promotion stage. Possible values are:</p> <ul style="list-style-type: none"> <li>Promotion Launched</li> </ul>

## PromotionStageEmailTemplate

Represents a junction between a promotion stage and an email template. With Global Promotions Management, a promotion's launch email is associated with the promotion's Promotion Launched stage. You can customize the promotion launch email based on your company's branding. 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

PromotionStageEmailTemplate is only available in orgs where Global Promotions Management is enabled.

## Fields

Field	Details
EmailCustomConfiguration	<p><b>Type</b> textarea</p> <p><b>Properties</b> Create, Nillable, Update</p> <p><b>Description</b> The customized content of an email template.</p>
EmailTemplateId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The ID of the email template associated with the promotion stage. This field is a relationship field.</p> <p><b>Relationship Name</b> EmailTemplate</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> EmailTemplate</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p>

Field	Details
	<p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p><b>Description</b> The unique number of the promotion stage email template.</p>
PromotionStageId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort</p> <p><b>Description</b> The ID of the promotion stage that the associated email template belongs to. This field is a relationship field.</p> <p><b>Relationship Name</b> PromotionStage</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> PromotionStage</p>

## TransactionJournal

Represents information about loyalty transactions that track activities, member behavior, and adjustments. This object is available in API version 51.0 and later.



**Warning:** You can add up to 50 custom fields in the TransactionJournal object.

## Supported Calls

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



## Fields

Field	Details
ActivityDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Create, Filter, Sort, Update</p> <p><b>Description</b> Required. The date of the transaction.</p>
BookedFareClass	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The class of ticket service that the member booked.</p>
BookedRoomType	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The type of room the member booked.</p>
BookingDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The date of the member booked a reservation or purchased goods or services.</p>
Brand	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The hotel chain where the member stayed or the brand of the purchased or rented goods or services.</p>
Channel	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p>

Field	Details
	<p><b>Description</b> The channel the member used for transactions.</p>
CurrencyIsoCode	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p> <p><b>Description</b> The three-letter ISO 4217 currency code of the monetary currency in which the transaction took place.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>• USD</li> </ul> <p>The default value is USD.</p>
Destination	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The ending location for a member's trip or the place where an order shipped to.</p>
Distance	<p><b>Type</b> double</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The miles or kilometers traveled by a member or an order's shipping distance.</p>
EndDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The end date associated with the transaction.</p>
EngagementChannelTypeId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p>

Field	Details
	<p><b>Description</b> The ID of the engagement channel that the customer used for the purchase. This field is available in API version 61.0 and later.</p> <p>This field is a relationship field.</p> <p><b>Relationship Name</b> EngagementChannelType</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> EngagementChannelType</p>
ErrorDescription	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The description of the error.</p>
EscrowPointsCreditDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The date on which the escrow points are credited to the associated loyalty program member's point balance. This field is available in API version 53.0 and later.</p>
Establishment	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The unique identifier for the location where the member stayed or purchased goods or services.</p>
ExternalTransactionNumber	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Nillable, Sort, Update</p> <p><b>Description</b> The transaction number from the external systems.</p>

Field	Details
FlightNumber	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The airline flight number the member booked.</p>
Industry	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> Specifies the type of the industry.</p>
InvoiceDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The date the invoice was generated.</p>
IsParentTransaction	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> Indicates whether the transaction journal represents the primary activity within a group of activities. This field is available in API version 60.0 and later.  The default value is <code>false</code>.</p>
JournalDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The date of the journal.</p>
JournalReason	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p>

Field	Details
	<p><b>Description</b></p> <p>Specifies the reason for the action that created a transaction journal. This field can store up to 999 values. After the field reaches 999 values, rename existing values that you no longer use instead of deleting values and adding new ones.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>• Accrual Cancellation Requested by Member</li> <li>• Order Canceled</li> <li>• Product Returned</li> <li>• Product or Service Unavailable</li> <li>• Redemption Cancellation Requested by Member</li> <li>• Resolve Disputed Points</li> <li>• Service Recovery</li> </ul>
JournalSubTypeId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The subtype of journal transaction. This is a relationship field.</p> <p><b>Relationship Name</b> JournalSubType</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> JournalSubType</p>
JournalTypeId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The type of journal transaction. This is a relationship field.</p> <p><b>Relationship Name</b> JournalType</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> JournalType</p>

Field	Details
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
LoyaltyProgramId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The loyalty program associated with the transaction. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgram</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgram</p>
MemberId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> The member associated with the transaction. This field is a polymorphic relationship field.</p> <p><b>Relationship Name</b> Member</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramMember</p>

Field	Details
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p><b>Description</b> The autogenerated number of the transaction journal.</p>
OrderId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The order associated with the transaction. This is a relationship field.</p> <p><b>Relationship Name</b> Order</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Order</p>
OrderItemId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The order items associated with the transaction. This is a relationship field.</p> <p><b>Relationship Name</b> OrderItem</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> OrderItem</p>
Origin	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The starting location for a member's trip or the place where an order shipped from.</p>

Field	Details
PartnerId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> Participating partners where members can add or redeem points. This is a relationship field.</p> <p><b>Relationship Name</b> Partner</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramPartner</p>
PaymentMethod	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The member's payment method.</p>
ProductCategoryId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The category of the product associated with the transaction. This is a relationship field.</p> <p><b>Relationship Name</b> ProductCategory</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> ProductCategory</p>
ProductId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The product associated with the transaction.</p>



Field	Details
	<p>This is a relationship field.</p> <p><b>Relationship Name</b> Product</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Product2</p>
PromotionId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The promotion associated with the transaction journal. This field is available in API version 52.0 and later. This is a relationship field.</p> <p><b>Relationship Name</b> Promotion</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Promotion</p>
Quantity	<p><b>Type</b> double</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The length of the booked stay or the order quantity.</p>
QuantityUnitOfMeasureId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The unit of measure for the booked stay or the order quantity. This is a relationship field.</p> <p><b>Relationship Name</b> QuantityUnitOfMeasure</p> <p><b>Relationship Type</b> Lookup</p>

Field	Details
	<p><b>Refers To</b> UnitOfMeasure</p>
RedeemedPointsExpirationInformation	<p><b>Type</b> textarea</p> <p><b>Properties</b> Create, Nillable, Update</p> <p><b>Description</b> The details of the fixed type non-qualifying points that are redeemed in a transaction journal and their corresponding expiration dates. This field is available in API version 55.0 and later.  RedeemedPointsExpirationInformation isn't able in the Object Manager and can't be customized.</p>
ReferredMemberId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The name of the person the member referred.  This is a relationship field.</p> <p><b>Relationship Name</b> ReferredMember</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramMember</p>
ReferredPartyId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The ID of the contact or account associated with the transaction journal. The contact ID belongs to the person whose activity the transaction journal represents. This field is available in API version 60.0 and later.</p> <p><b>Relationship Name</b> ReferredParty</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Account, Contact</p>

Field	Details
RelatedCorporateMembershipId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The corporate membership associated with the transaction. This is a relationship field.</p> <p><b>Relationship Name</b> RelatedCorporateMembership</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramMember</p>
RelatedJournalId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The related journal entry associated with this transaction. This is a relationship field.</p> <p><b>Relationship Name</b> RelatedJournal</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> TransactionJournal</p>
StartDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The start date associated with the transaction.</p>
Status	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p> <p><b>Description</b> Required. The status of the transaction.</p>

Field	Details
	<p>Possible values are:</p> <ul style="list-style-type: none"> <li>• Cancelled</li> <li>• Error</li> <li>• Pending</li> <li>• Processed</li> </ul>
TotalPromotionDiscount	<p><b>Type</b> currency</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The total discount provided by the associated promotion for the activity that the transaction journal represents. This field is available in API version 61.0 and later.</p>
TransactionAmount	<p><b>Type</b> currency</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> For reservations, the total fees or prepaid balance. For goods and services, the total purchase price.</p>
TransactionLocation	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The location where the loyalty member purchased goods or services.</p>
TransactionSourceLocationId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The ID of the retail store where the activity that the transaction journal represents has taken place. This field is available in API version 60.0 and later. This field is a relationship field.</p> <p><b>Relationship Name</b> TransactionSourceLocation</p> <p><b>Relationship Type</b> Lookup</p>

Field	Details
	<b>Refers To</b> RetailStore
TraveledFareClass	<b>Type</b> picklist <b>Properties</b> Create, Filter, Group, Nillable, Sort, Update <b>Description</b> The class of ticket service that the member traveled.
UsageType	<b>Type</b> picklist <b>Properties</b> Create, Filter, Group, Nillable, Restricted picklist, Sort <b>Description</b> The type of usage. Possible values are: <ul style="list-style-type: none"> <li>• Loyalty</li> <li>• Rebates</li> <li>• UnifiedPromotions—This value is available in API version 60.0 and later.</li> </ul>
VoucherCode	<b>Type</b> string <b>Properties</b> Create, Filter, Group, Nillable, Sort, Update <b>Description</b> The voucher code generated by an external system.

## Usage

You can query Transaction Journal records in your org using a combination of these indexed fields:

- ActivityDate
- ExternalTransactionNumber
- Id
- JournalTypeId
- LoyaltyProgramId
- MemberId
- OrderId
- RelatedJournalId
- UsageType

The TransactionJournal object in most orgs can store millions of records. A limited set of TransactionJournal fields are indexed to help optimize operations on the TransactionJournal 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.

**TransactionJournalChangeEvent** on page 217 (API Version 63.0)

Change events are available for the object.

## Voucher

---

Represents details about a voucher that's issued to a loyalty program member. This object is available in API version 51.0 and later.

## Supported Calls

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

## Fields

Field	Details
AccountId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> The ID of the account to which the voucher is issued. This field is available in API version 62.0 and later.  If you want to issue vouchers to accounts, set up a custom process based on the requirements of your company.  This field is a relationship field.</p> <p><b>Relationship Name</b> Account</p> <p><b>Refers To</b> Account</p>
BenefitId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p>

Field	Details
	<p><b>Description</b> The benefit associated with the voucher. This is a relationship field.</p> <p><b>Relationship Name</b> Benefit</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Benefit</p>
ContactId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> The contact associated with the member. This is a relationship field.</p> <p><b>Relationship Name</b> Contact</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Contact</p>
DiscountPercent	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The discount percentage associated with the voucher.</p>
EffectiveDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The date the voucher goes into effect.</p>
EffectiveDateTime	<p><b>Type</b> dateTime</p>

Field	Details
	<p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The date and time from when the voucher is available for redemption. This field is available in API version 62.0 and later.</p>
ExpirationDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The date the voucher expires.</p>
ExpirationDateTime	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The date and time when the voucher expires and is no longer available for redemption. This field is available in API version 62.0 and later.</p>
FaceValue	<p><b>Type</b> currency</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The value of the voucher.</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>



Field	Details
LoyaltyProgramMemberId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> The member who has been issued the voucher. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgramMember</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgramMember</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p><b>Description</b> The autogenerated number of the voucher.</p>
OwnerId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> The ID of the user who owns this record. This is a polymorphic relationship field.</p> <p><b>Relationship Name</b> Owner</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Group, User</p>
PromotionId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The promotion associated with the voucher.</p>

Field	Details
	<p>This is a relationship field.</p> <p><b>Relationship Name</b> Promotion</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Promotion</p>
Reason	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The reason why the voucher was issued.</p>
RedeemedValue	<p><b>Type</b> currency</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The voucher balance that was redeemed. This field is available in API version 53.0 and later.</p>
RemainingValue	<p><b>Type</b> currency</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The remaining value of the voucher after partial redemption. This field is available in API version 53.0 and later.</p>
ReservationKey	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The key that's used to redeem a reserved voucher. This field is available in API version 62.0 and later.</p>
ReservedDateTime	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p>

Field	Details
	<p><b>Description</b></p> <p>The date and time when the voucher is reserved for redemption. This field is available in API version 62.0 and later.</p>
ReservedValue	<p><b>Type</b></p> <p>currency</p> <p><b>Properties</b></p> <p>Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b></p> <p>The value of the voucher reserved by the customer. This field is available in API version 62.0 and later.</p>
Status	<p><b>Type</b></p> <p>picklist</p> <p><b>Properties</b></p> <p>Create, Defaulted on create, Filter, Group, Restricted picklist, Sort, Update</p> <p><b>Description</b></p> <p>Required. The status of the voucher.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>• Cancelled</li> <li>• Expired</li> <li>• Issued</li> <li>• Redeemed</li> </ul>
TransactionJournalId	<p><b>Type</b></p> <p>reference</p> <p><b>Properties</b></p> <p>Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b></p> <p>The transaction journal associated with the member voucher when the voucher results from accrual of points.</p> <p>This is a relationship field.</p> <p><b>Relationship Name</b></p> <p>TransactionJournal</p> <p><b>Relationship Type</b></p> <p>Lookup</p> <p><b>Refers To</b></p> <p>TransactionJournal</p>
UseDate	<p><b>Type</b></p> <p>date</p>

Field	Details
	<p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The date the voucher was used.</p>
VoucherCode	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The voucher identifier generated by an external system.</p>
VoucherDefinitionId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort</p> <p><b>Description</b> The loyalty voucher definition associated with the voucher. This is a relationship field.</p> <p><b>Relationship Name</b> VoucherDefinition</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> VoucherDefinition</p>

## 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.

**[VoucherChangeEvent](#) on page 217 (API Version 63.0)**

Change events are available for the object.

**[VoucherHistory](#) (API version 52.0)**

History is available for tracked fields of the object.

**[VoucherShare](#) (API version 52.0)**

Sharing is available for the object.

## VoucherDefinition

---

Represents details about a voucher definition that's associated with a loyalty program. This object is available in API version 51.0 and later.

### Supported Calls

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

### Fields

Field	Details
<code>AutogeneratedCodeLength</code>	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The length of the auto-generated voucher code. If no value is specified, the default is eight digits. This field is available in API version 57.0 and later.</p>
<code>CodePrefix</code>	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The prefix of the auto-generated voucher code. This field is available in API version 57.0 and later.</p>
<code>CodeSuffix</code>	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The suffix of the auto-generated voucher code. This field is available in API version 57.0 and later.</p>
<code>CostPerVoucher</code>	<p><b>Type</b> currency</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p>

Field	Details
	<p><b>Description</b> The liability incurred for each voucher that's issued. This field is available in API version 52.0 and later.</p>
Description	<p><b>Type</b> textarea</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The description of the voucher.</p>
DiscountPercent	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The discount percentage associated with the voucher.</p>
EffectiveDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. The date on which the voucher becomes valid.</p>
EffectiveDateTime	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The date and time from when the vouchers issued by the voucher definition can be redeemed by customers. This field is available in API version 62.0 and later.</p>
ExpirationDate	<p><b>Type</b> date</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The date the voucher expires. The expiration date is only valid for the fixed date model.</p>
ExpirationDateTime	<p><b>Type</b> dateTime</p>

Field	Details
	<p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The date and time on which the vouchers issued by the voucher definition are no longer available for customers to redeem. This field is available in API version 62.0 and later.</p>
ExpirationPeriod	<p><b>Type</b> int</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The number of days or months in the expiration period.</p>
ExpirationPeriodUnit	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p><b>Description</b> The unit for the expiration period. Possible values are:</p> <ul style="list-style-type: none"> <li>• Days</li> <li>• Months</li> </ul>
ExpirationType	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Restricted picklist, Sort, Update</p> <p><b>Description</b> Required. The type of expiration for the voucher. Possible values are:</p> <ul style="list-style-type: none"> <li>• FixedDate</li> <li>• Period</li> </ul>
FaceValue	<p><b>Type</b> currency</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The value of the voucher.</p>

Field	Details
HasTimeBasedVoucherPeriod	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort</p> <p><b>Description</b> Indicates whether vouchers issued using the definition are effective and expire at a specific date and time. This field is available in API version 62.0 and later.  The default value is <code>false</code>.</p>
ImageUrl	<p><b>Type</b> url</p> <p><b>Properties</b> Create, Filter, Nillable, Sort, Update</p> <p><b>Description</b> The URL for the voucher definition image. This field is available in API version 57.0 and later.</p>
IsActive	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> Required. Indicates whether the voucher is active (<code>true</code>) or not (<code>false</code>).  The default value is <code>false</code>.</p>
IsPartialRedeemable	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> Indicates whether the voucher can be partially redeemed (<code>true</code>) or not (<code>false</code>). This field is available in API version 53.0 and later.  The default value is <code>false</code>.</p>
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user referenced this record.</p>



Field	Details
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The most recent date on which a user viewed this record.</p>
LoyaltyProgramId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The ID of the loyalty program. This is a relationship field.</p> <p><b>Relationship Name</b> LoyaltyProgram</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> LoyaltyProgram</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, idLookup, Sort, Update</p> <p><b>Description</b> The name of the voucher.</p>
OwnerId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> The ID of the user who owns this record. This is a polymorphic relationship field.</p> <p><b>Relationship Name</b> Owner</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Group, User</p>

Field	Details
PartnerAccountId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The partner associated with the voucher. This is a relationship field.</p> <p><b>Relationship Name</b> PartnerAccount</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Account</p>
ProductCategoryDerivedFieldId	<p><b>Type</b> reference</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> A look up to show product category in voucher definition. This is a relationship field.</p> <p><b>Relationship Name</b> ProductCategoryDerivedField</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> ProductCategory</p>
ProductDerivedFieldId	<p><b>Type</b> reference</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> A look up to show product in voucher definition. This is a relationship field.</p> <p><b>Relationship Name</b> ProductDerivedField</p> <p><b>Relationship Type</b> Lookup</p>

Field	Details
	<p><b>Refers To</b> Product2</p>
ProductId	<p><b>Type</b> reference</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The product associated with the voucher. This is a polymorphic relationship field.</p> <p><b>Relationship Name</b> Product</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Product2, ProductCategory</p>
Type	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Restricted picklist, Sort, Update</p> <p><b>Description</b> Required. The type of voucher. Possible values are:</p> <ul style="list-style-type: none"> <li>• <code>DiscountPercentage</code>—Discount Percentage</li> <li>• <code>FixedValue</code>—Fixed Value</li> <li>• <code>ProductOrService</code>—Product or Service</li> </ul>

## 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.

### [VoucherDefinitionChangeEvent](#) on page 217 (API Version 62.0)

Change events are available for the object.

### [VoucherDefinitionHistory](#)

History is available for tracked fields of the object.

### [VoucherDefinitionShare](#)

Sharing is available for the object.

## CHAPTER 2 Loyalty Management Fields on Standard Objects

### In this chapter ...

- [Loyalty Management Fields on EngagementChannelType](#)
- [Loyalty Management Fields on InternalOrganizationUnit](#)

Loyalty Management has added standard fields to some standard Salesforce objects to represent information specific to a promotion. These fields are available only in orgs where Global Promotions Management is enabled.

## Loyalty Management Fields on EngagementChannelType

---

Standard fields extend the EngagementChannelType object for use in Loyalty Management to represent to information promotion eligibility criteria. The Loyalty Management Fields on EngagementChannelType are 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
Delivery	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p><b>Description</b> The type of channel that's eligible for a promotion. This field is only available in orgs where Global Promotions Management is enabled.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>• Online</li> <li>• Social</li> </ul>

## Loyalty Management Fields on InternalOrganizationUnit

---

Standard fields extend the InternalOrganizationUnit object for use in Loyalty Management to represent to information about Global Promotion Management's connection with a Marketing Cloud business unit. The Loyalty Management fields on InternalOrganizationUnit are 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

The Delivery field is only available in orgs where Global Promotions Management is enabled.

## Fields

Field	Details
BusinessUsage	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p><b>Description</b> The business purpose of the record. Possible values are:</p> <ul style="list-style-type: none"> <li>• UNIFIED_PROMOTIONS—Global Promotions Management</li> </ul>
EmailConfiguration	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The send classification that's used by the connected Marketing Cloud business unit to send emails.</p>
OrganizationIdentifier	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Filter, Group, Nillable, Sort, Update</p> <p><b>Description</b> The name of the company's Marketing Cloud business unit that Global Promotions Management is connected to.</p>

## CHAPTER 3 Coupon Objects in Loyalty Management

Global Promotions Management in Loyalty Management provides access to Coupon objects that you can use to create coupons and manage coupon redemption for your company's promotions. When Global Promotions Management is enabled, you don't need any additional license or user permission for these coupon objects.

For more information about the objects you can use to manage coupons for promotions:

- [Coupon](#)
- [CouponCodeRedemption](#)

## CHAPTER 4 Loyalty Management Associated Objects

In this chapter ...

- [Standard Name Change](#)

This section provides a list of objects associated to standard objects and their standard fields.



## StandardObjectNameChangeEvent

---

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",
    "CreatedBy": "<User_ID>",
    "LastModifiedDate": "2021-02-09T23:17:59Z",
    "LastModifiedBy": "<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](#).

## Objects That Support Change Events

These objects have associated `ChangeEvent` objects.

- BenefitType
- GameParticipant
- GameParticipantReward
- GameReward
- JournalSubType
- JournalType
- LoyaltyMemberCurrency
- LoyaltyMemberTier
- LoyaltyPartnerProduct
- LoyaltyPgmEngmtAttribute
- LoyaltyPgmEngmtAttrProm
- LoyaltyPgmGroupMbrRInsp
- LoyaltyPgmMbrAttributeVal
- LoyaltyPgmPartnerPromotion
- LoyaltyProgram
- LoyaltyProgramBadge
- LoyaltyProgramCurrency
- LoyaltyProgramMbrPromotion
- LoyaltyProgramMember
- LoyaltyProgramMemberBadge
- LoyaltyProgramMemberCase
- LoyaltyProgramPartner
- LoyaltyTier
- LoyaltyTierBenefit
- LoyaltyTierGroup
- MemberBenefit
- Promotion
- PromotionLoyaltyPtrnProdt
- PromotionMarketSegment
- PromotionPartyUsage
- TransactionJournal
- Voucher
- VoucherDefinition

## CHAPTER 5 Apex Reference

### In this chapter ...

- [ConnectApi Namespace](#)
- [LoyaltyManagement Namespace](#)

This Apex reference goes into detail about the built-in Apex `ConnectApi` classes and interfaces.

For Apex-related objects and SOAP API calls, see [Objects and SOAP API Calls and Headers for Apex](#) in the *SOAP API Developer Guide*.

## ConnectApi Namespace

---

The `ConnectApi` namespace (also called Connect in Apex) provides classes for accessing the same data available in Connect REST API. Use Connect in Apex to create custom experiences in Salesforce.

For information about working with the `ConnectApi` classes, see [Connect in Apex](#).

### [LoyaltyConnect Methods](#)

This is a method for `LoyaltyConnect`. Method is static.

### [LoyaltyManagementConnect Methods](#)

This is a method for `LoyaltyManagementConnect`. Method is static.

### [LoyaltyVoucherConnect Methods](#)

This is a method for `LoyaltyVoucherConnect`. Method is static.

### [Input Classes](#)

These are Loyalty Management Apex input classes.

### [Output Classes](#)

These are Loyalty Management Apex output classes.

### [ConnectApi Enums](#)

Enums specific to the `ConnectApi` namespace.

## LoyaltyConnect Methods

This is a method for `LoyaltyConnect`. Method is static.

### [getMemberBenefits\(memberId\)](#)

Get benefits for a loyalty program member.

### **getMemberBenefits (memberId)**

Get benefits for a loyalty program member.

### API Version

51.0

### Requires Chatter

No

### Signature

```
public static ConnectApi.MemberBenefitsOutput getMemberBenefits(String memberId)
```

## Parameters

*memberId*

Type: String

ID of the loyalty program member.

## Return Value

Type: [ConnectApi.MemberBenefitsOutput](#)

# LoyaltyManagementConnect Methods

This is a method for `LoyaltyManagementConnect`. Method is static.

[enrollCorporateMember\(loyaltyProgramName, memberDetails\)](#)

Enroll corporate members in a loyalty program.

[enrollIndividualMember\(loyaltyProgramName, memberDetails\)](#)

Enroll individual members in a loyalty program.

[getMemberProfileUsingLoyaltyProgramMemberId\(loyaltyProgramName, memberId, programCurrencyName\)](#)

Get the details of a loyalty program member using the program member ID.

[getMemberProfileUsingMembershipNumber\(loyaltyProgramName, membershipNumber, programCurrencyName\)](#)

Get the details of a loyalty program member using the program membership number.

## **enrollCorporateMember(loyaltyProgramName, memberDetails)**

Enroll corporate members in a loyalty program.

## API Version

55.0

## Requires Chatter

No

## Signature

```
public static ConnectApi.CorporateMemberOutput enrollCorporateMember(String
loyaltyProgramName, ConnectApi.CorporateMemberInput memberDetails)
```

## Parameters

*loyaltyProgramName*

Type: String

The name of the loyalty program in which the member is enrolled.

*memberDetails*

Type: [ConnectApi.CorporateMemberInput](#)

Input representation of the corporate member create request.

## Return Value

Type: [ConnectApi.CorporateMemberOutput](#)

## Example

```
public class TestCorporateEnrollment {
    Map<String, String> additionalDetails = new Map<String, String>();
    ConnectApi.MemberAccountInput account = new ConnectApi.MemberAccountInput();
    additionalDetails.put('BillingCity', 'Delhi');
    account.additionalAccountFieldValues = additionalDetails;
    account.name = 'BBBB';
    ConnectApi.CorporateMemberInput member = new ConnectApi.CorporateMemberInput();
    member.associatedAccountDetails = account;
    DateTime myDateTime = DateTime.newInstance(2021, 1, 31, 7, 8, 16);
    member.enrollmentDate = myDateTime;
    member.membershipNumber = 'abcdefgh';
    member.memberStatus = 'Active';
    ConnectApi.CorporateMemberOutput output =
    ConnectApi.LoyaltyManagementConnect.enrollCorporateMember('Program-999',member);
}
```

## **enrollIndividualMember(loyaltyProgramName, memberDetails)**

Enroll individual members in a loyalty program.

## API Version

55.0

## Requires Chatter

No

## Signature

```
public static ConnectApi.IndividualMemberOutput enrollIndividualMember(String
loyaltyProgramName, ConnectApi.IndividualMemberInput memberDetails)
```

## Parameters

*loyaltyProgramName*

Type: String

The name of the loyalty program in which the member is enrolled.

*memberDetails*

Type: [ConnectApi.IndividualMemberInput](#)

Input representation of the individual member create request.

## Return Value

Type: `ConnectApi.IndividualMemberOutput`

## Example

```
public class TestIndividualEnrollment {
    Map<String, String> additionalDetails = new Map<String, String>();
    ConnectApi.MemberContactInput contact = new ConnectApi.MemberContactInput();
    additionalDetails.put('PersonMailingCity', 'Delhi');
    contact.additionalContactFieldValues = additionalDetails;
    contact.lastName = 'BBBB';
    ConnectApi.IndividualMemberInput member = new ConnectApi.IndividualMemberInput();
    member.associatedContactDetails = contact;
    DateTime myDateTime = DateTime.newInstance(2021, 1, 31, 7, 8, 16);
    member.enrollmentDate = myDateTime;
    member.membershipNumber = 'abcdefg';
    member.memberStatus = 'Active';
    ConnectApi.IndividualMemberOutput output =
    ConnectApi.LoyaltyManagementConnect.enrollIndividualMember('Program1', member);
}
```

## **getMemberProfileUsingLoyaltyProgramMemberId(loyaltyProgramName, memberId, programCurrencyName)**

Get the details of a loyalty program member using the program member ID.

### API Version

55.0

### Requires Chatter

No

### Signature

```
public static ConnectApi.MemberDetailsOutput
getMemberProfileUsingLoyaltyProgramMemberId(String loyaltyProgramName, String memberId,
String programCurrencyName)
```

### Parameters

*loyaltyProgramName*

Type: String

The name of the loyalty program.

*memberId*

Type: String

The ID of the loyalty program member.The ID of the loyalty program member.



*programCurrencyName*

Type: String

The name of the loyalty program currency associated with the member.

Use this parameter to get the details of the member's points-related information for a specific currency.

## Return Value

Type: `ConnectApi.MemberDetailsOutput`

## Example

```
public class TestMemberId {
    ConnectApi.MemberDetailsOutput details =
    ConnectApi.LoyaltyManagementConnect.getMemberProfileUsingLoyaltyProgramMemberId('program1',
    '01Mxx00000007X1EAI', 'ruble');
}
```

## **getMemberProfileUsingMembershipNumber(loyaltyProgramName, membershipNumber, programCurrencyName)**

Get the details of a loyalty program member using the program membership number.

## API Version

55.0

## Requires Chatter

No

## Signature

```
public static ConnectApi.MemberDetailsOutput getMemberProfileUsingMembershipNumber(String
loyaltyProgramName, String membershipNumber, String programCurrencyName)
```

## Parameters

*loyaltyProgramName*

Type: String

The name of the loyalty program.

*membershipNumber*

Type: String

The membership number of the loyalty program member.

*programCurrencyName*

Type: String

The name of the loyalty program currency associated with the member.

Use this parameter to get the details of the member's points-related information for a specific currency.

## Return Value

Type: `ConnectApi.MemberDetailsOutput`

## Example

```
public class TestMembershipNumber {
    ConnectApi.MemberDetailsOutput details =
    ConnectApi.LoyaltyManagementConnect.getMemberProfileUsingMembershipNumber('program1',
    'Member15', 'ruble');
}
```

## LoyaltyVoucherConnect Methods

This is a method for `LoyaltyVoucherConnect`. Method is static.

[redeem\(programName, membershipNumber, voucherCode\)](#)

Redeem a voucher for a program member. The status of the voucher gets updated to redeemed and the use date gets updated to the current date.

[redeem\(programName, membershipNumber, voucherCode, amount\)](#)

Redeem partial value of a voucher for a program member. The status of the voucher gets updated to redeemed and the use date gets updated to the current date.

[redeem\(programName, membershipNumber, voucherCode, voucherNumber\)](#)

Redeem the single-use voucher for a loyalty program member. The status of the voucher gets updated to redeemed and the use date gets updated to the current date.

[redeem\(programName, membershipNumber, voucherCode, voucherNumber, amount\)](#)

Redeem partial value of a voucher for a program member. The status of the voucher gets updated to redeemed and the use date gets updated to the current date.

### **redeem(programName, membershipNumber, voucherCode)**

Redeem a voucher for a program member. The status of the voucher gets updated to redeemed and the use date gets updated to the current date.

## API Version

51.0

## Requires Chatter

Yes

## Signature

```
public static ConnectApi.RedeemVoucherOutput redeem(String programName, String membershipNumber, String voucherCode)
```

## Parameters

*programName*

Type: String

The loyalty program name.

*membershipNumber*

Type: String

The loyalty program membership number.

*voucherCode*

Type: String

The voucher code to redeem.



**Note:** The voucher definition associated with the voucher code must be active to redeem the voucher.

## Status Code

Returns HTTP 201 on success.

## **redeem(programName, membershipNumber, voucherCode, amount)**

Redeem partial value of a voucher for a program member. The status of the voucher gets updated to redeemed and the use date gets updated to the current date.

## API Version

53.0

## Requires Chatter

Yes

## Signature

```
public static ConnectApi.RedeemVoucherOutput redeem(String programName, String membershipNumber, String voucherCode, ConnectApi.RedeemVoucherInput amount)
```

## Parameters

*programName*

Type: String

The loyalty program name.

*membershipNumber*

Type: String

The loyalty program membership number.

*voucherCode*

Type: String

The voucher code to redeem.



**Note:** The voucher definition associated with the voucher code must be active to redeem the voucher.

*amount*

Type: [ConnectApi.RedeemVoucherInput](#) on page 235

The value of the voucher to be redeemed.

## Return Value

Type: [ConnectApi.RedeemVoucherOutput](#) on page 236

## Example

```
public class TestRedeemVoucher {
    ConnectApi.RedeemVoucherInput redeemInput = new ConnectApi.RedeemVoucherInput();
    redeemInput.redeemAmount = 10.2;
    ConnectApi.RedeemVoucherOutput output =
ConnectApi.LoyaltyVoucherConnect.redeem('DemoProgram', 'M1234', 'VC001', redeemInput);
}
```

## **redeem(programName, membershipNumber, voucherCode, voucherNumber)**

Redeem the single-use voucher for a loyalty program member. The status of the voucher gets updated to redeemed and the use date gets updated to the current date.

## API Version

55.0

## Requires Chatter

No

## Signature

```
public static ConnectApi.RedeemVoucherOutput redeem(String programName, String
membershipNumber, String voucherCode, String voucherNumber)
```

## Parameters

*programName*

Type: String

Description

*membershipNumber*

Type: String

Description

*voucherCode*

Type: String

Description

*voucherNumber*

Type: String

Description

## Return Value

Type: [ConnectApi.RedeemVoucherOutput](#)

## Example

```
public class TestRedeemVoucher {
    ConnectApi.RedeemVoucherOutput output =
ConnectApi.LoyaltyVoucherConnect.redeem('DemoProgram', 'M1234', 'VC001', '00000006');
}
```

## **redeem(programName, membershipNumber, voucherCode, voucherNumber, amount)**

Redeem partial value of a voucher for a program member. The status of the voucher gets updated to redeemed and the use date gets updated to the current date.

## API Version

55.0

## Requires Chatter

No

## Signature

```
public static ConnectApi.RedeemVoucherOutput redeem(String programName, String
membershipNumber, String voucherCode, String voucherNumber, ConnectApi.RedeemVoucherInput
amount)
```

## Parameters

*programName*

Type: String

The loyalty program name.

*membershipNumber*

Type: String

The loyalty program membership number.

*voucherCode*

Type: String

The voucher code to redeem.



**Note:** The voucher definition associated with the voucher code must be active to redeem the voucher.

*voucherNumber*

Type: String

The auto-generated voucher number.

*amount*

Type: [ConnectApi.RedeemVoucherInput](#)

The value of the voucher to be redeemed.

## Return Value

Type: [ConnectApi.RedeemVoucherOutput](#)

## Example

```
public class TestRedeemVoucher {
    ConnectApi.RedeemVoucherInput redeemInput = new ConnectApi.RedeemVoucherInput();
    redeemInput.redeemAmount = 10.2;
    ConnectApi.RedeemVoucherOutput output =
ConnectApi.LoyaltyVoucherConnect.redeem('DemoProgram', 'M1234', 'VC001', '00000006',
redeemInput);
}
```

## Input Classes

These are Loyalty Management Apex input classes.

[ConnectApi.CorporateMemberInput](#)

Input representation of the corporate member create request.

[ConnectApi.IndividualMemberInput](#)

Input representation of the individual member create request.

[ConnectApi.MemberAccountInput](#)

Input representation of the loyalty member account.

[ConnectApi.MemberContactInput](#)

Input representation of the loyalty member contact.

[ConnectApi.RedeemVoucherInput](#)

Input representation to redeem amount from a voucher.

## ConnectApi.CorporateMemberInput

Input representation of the corporate member create request.

Property	Type	Description	Required or Optional	Available Version
additionalMemberFieldValues	Map<String,String>	The custom attributes of the member in the loyalty program member record. <ul style="list-style-type: none"> <li>Keys</li> <li>Values</li> </ul>	Optional	55.0
associatedAccountDetails	<a href="#">ConnectApi.LoyaltyProgramDetails</a>	The account details of the member.	Required	55.0
canReceivePartnerPromotions	Boolean	Indicates whether the member can receive partner promotions for the program ( <code>true</code> ) or not ( <code>false</code> ).	Optional	55.0
canReceivePromotions	Boolean	Indicates whether the member can receive promotions for the program ( <code>true</code> ) or not ( <code>false</code> ).	Optional	55.0
createTransactionJournals	Boolean	Indicates whether to create the transaction journal records ( <code>true</code> ) or not ( <code>false</code> ). The transaction journals records are created to: <ul style="list-style-type: none"> <li>Credit points to the member for enrolling in the loyalty program.</li> <li>Credit points to the member who has referred the corporate member.</li> </ul>	Optional	55.0
enrollmentChannel	<a href="#">ConnectApi.EnrollmentChannelResource</a>	The channel used by members to enroll into the loyalty program. Possible values are: <ul style="list-style-type: none"> <li>CallCenter</li> <li>Email</li> <li>Franchise</li> <li>Mobile</li> <li>Partner</li> <li>Pos</li> <li>Print</li> <li>Social</li> <li>Store</li> <li>Web</li> </ul>	Optional	55.0

Property	Type	Description	Required or Optional	Available Version
enrollmentDate	Datetime	The date on which a member is enrolled. The format is: YYYY-MM-DDTHH:MM:SS.	Required	55.0
memberStatus	String	The status of the member.  Possible values are: <ul style="list-style-type: none"> <li>Active</li> <li>Inactive</li> <li>Custom value</li> </ul>	Required	55.0
membershipEndDate	Datetime	The date on which the loyalty program member's membership ends.	Optional	55.0
membershipNumber	String	The unique membership number of the loyalty program member.	Required	55.0
referredBy	String	The membership number of the loyalty program member who has referred the member.	Optional	55.0
transactionJournalStatementFrequency	ConnectApi.StatementFrequencyResource	The frequency at which transaction journal statement must be delivered to the member.  Possible values are: <ul style="list-style-type: none"> <li>Monthly</li> <li>Quarterly</li> </ul>	Optional	55.0
transactionJournalStatementMethod	ConnectApi.StatementMethodResource	The method used to deliver transaction journal statements to the member.  Possible values are: <ul style="list-style-type: none"> <li>Email</li> <li>Mail</li> </ul>	Optional	55.0

## ConnectApi.IndividualMemberInput

Input representation of the individual member create request.

Property	Type	Description	Required or Optional	Available Version
additionalMemberFieldValues	Map<String,String>	The custom field values in the loyalty program member record. <ul style="list-style-type: none"> <li>Keys</li> <li>Values</li> </ul>	Required	55.0



Property	Type	Description	Required or Optional	Available Version
associatedContactDetails	<a href="#">ContactDetails</a>	The contact record of the individual type member.	Required	55.0
canReceivePartnerPromotions	Boolean	Indicates whether the member can receive partner promotions for the program ( <code>true</code> ) or not ( <code>false</code> ).	Optional	55.0
canReceivePromotions	Boolean	Indicates whether the member can receive promotions for the program ( <code>true</code> ) or not ( <code>false</code> ).	Optional	55.0
createTransactionJournals	Boolean	Indicates whether to create the transaction journal records ( <code>true</code> ) or not ( <code>false</code> ).  The transaction journal records are created to: <ul style="list-style-type: none"> <li>• Credit points to the individual member for enrolling in the loyalty program.</li> <li>• Credit points to the member who has referred the individual member.</li> </ul>	Optional	55.0
enrollmentChannel	<a href="#">ConnectApi.EnrollmentChannelResource</a>	The channel used by members to enroll into the loyalty program.  Possible values are: <ul style="list-style-type: none"> <li>• CallCenter</li> <li>• Email</li> <li>• Franchise</li> <li>• Mobile</li> <li>• Partner</li> <li>• Pos</li> <li>• Print</li> <li>• Social</li> <li>• Store</li> <li>• Web</li> </ul>	Optional	55.0
enrollmentDate	Datetime	The date on which a member is enrolled. The format is: YYYY-MM-DDTHH:MM:SS.	Required	55.0
memberStatus	String	The status of the member.  Possible values are: <ul style="list-style-type: none"> <li>• Active</li> <li>• Inactive</li> <li>• Custom value</li> </ul>	Required	55.0

Property	Type	Description	Required or Optional	Available Version
membershipEndDate	Datetime	The date on which the loyalty program member's membership ends.	Optional	55.0
membershipNumber	String	The unique membership number of the loyalty program member.	Required	55.0
referredBy	String	The membership number of the loyalty program member who has referred the member.	Optional	55.0
relatedCorporateMembershipNumber	String	The external corporate membership of the member.	Optional	55.0
transactionJournalStatementFrequency	ConnectApi.StatementFrequencyResource	The frequency at which transaction journal statement must be delivered to the member.  Possible values are: <ul style="list-style-type: none"> <li>Monthly</li> <li>Quarterly</li> </ul>	Optional	55.0
transactionJournalStatementMethod	ConnectApi.StatementMethodResource	The method used to deliver transaction journal statements to the member.  Possible values are: <ul style="list-style-type: none"> <li>Email</li> <li>Mail</li> </ul>	Optional	55.0

## ConnectApi.MemberAccountInput

Input representation of the loyalty member account.

Property	Type	Description	Required or Optional	Available Version
additionalAccountFieldValues	Map<String,String>	The additional attribute values that must be saved in the account record.	Optional	55.0
allowDuplicateRecords	Boolean	Indicates whether a duplicate member record can be created when a record with identical details already exists ( <code>true</code> ) or not ( <code>false</code> ).	Optional	55.0
name	String	The name of the account.	Required	55.0
phone	String	The phone number associated with the account.	Optional	55.0
website	String	The website associated with the account.	Optional	55.0

## ConnectApi.MemberContactInput

Input representation of the loyalty member contact.

Property	Type	Description	Required or Optional	Available Version
additionalContactFieldValues	Map<String,String>	The additional attributes that must be saved in the person account.	Optional	55.0
allowDuplicateRecords	Boolean	Indicates whether a duplicate record can be created for a member when an existing person account record already exists ( <code>true</code> ) or not ( <code>false</code> ).	Optional	55.0
email	String	The email ID of the contact.	Optional	55.0
firstName	String	The first name of the contact.	Optional	55.0
lastName	String	The last name of the contact.	Required	55.0

## ConnectApi.RedeemVoucherInput

Input representation to redeem amount from a voucher.

Property	Type	Description	Required or Optional	Available Version
currencyIsoCode	String	Three-letter ISO 4217 currency code of the monetary currency in which the voucher is redeemed.	Optional	55.0
redeemAmount	Double	The value of the voucher to be redeemed. Specify this amount when the associated voucher definition is partially redeemable and is of the type Fixed Value.	Optional	53.0

## Output Classes

These are Loyalty Management Apex output classes.

### [ConnectApi.MemberBenefitsOutput](#)

Output representation of all benefits available for a member.

### [ConnectApi.MemberBenefitOutput](#)

Output representation of an individual benefit available for a member.

### [ConnectApi.RedeemVoucherOutput](#)

Output representation of a voucher redemption.

### [ConnectApi.CorporateMemberOutput](#)

Output representation of the corporate member enrollment create request.

[ConnectApi.IndividualMemberOutput](#)

Output representation of the individual member create request.

[ConnectApi.MemberAccountOutput](#)

Output representation of the details of a loyalty program member's associated account record.

[ConnectApi.MemberContactOutput](#)

Output representation of the details of a loyalty program member's associated contact record.

[ConnectApi.MemberDetailsOutput](#)

Output representation of the details of a loyalty program member.

[ConnectApi.MemberCurrencyOutput](#)

Output representation of the details of the loyalty program member's associated currencies.

[ConnectApi.MemberTierOutput](#)

Output representation of the details of the loyalty program member's associated tiers.

[ConnectApi.TransactionJournalOutput](#)

Output representation of the transaction journal records.

## ConnectApi.MemberBenefitsOutput

Output representation of all benefits available for a member.

Property Name	Type	Description	Available Version
memberBenefits	List< <a href="#">ConnectApi.MemberBenefitOutput</a> >	List of all benefits available for a member.	51.0



## ConnectApi.MemberBenefitOutput

Output representation of an individual benefit available for a member.

Property Name	Type	Description	Available Version
benefitId	String	ID of the benefit available for the member.	51.0
benefitName	String	The name of the benefit available for the member.	51.0
benefitTypeId	String	ID of the associated benefit type record.	51.0
benefitTypeName	String	The name of the associated benefit type.	51.0
createdRecordId	String	ID of the record created during benefit assignment.	51.0
createdRecordName	String	The display name of the record created during benefit assignment.	51.0
isActive	Boolean	Indicates whether the benefit is active.	51.0

## ConnectApi.RedeemVoucherOutput

Output representation of a voucher redemption.

Property Name	Type	Description	Available Version
remainingAmount	Double	The voucher value after redemption.  <b>Note:</b> If the complete value of the voucher is redeemed, the remaining amount is zero.	53.0
currencyIsoCode	String	Three-letter ISO 4217 currency code of the remaining voucher value.  <b>Note:</b> If multicurrency isn't enabled, the currency code is set to the default currency code of an org.	56.0

## ConnectApi.CorporateMemberOutput

Output representation of the corporate member enrollment create request.

Property Name	Type	Description	Available Version
accountId	String	The ID of the account record associated with the loyalty program member record.	55.0
loyaltyProgramMemberId	String	The ID of the loyalty program member record created for the corporate member.	55.0
loyaltyProgramName	String	The name of the loyalty program in which the corporate member is enrolled.	55.0
membershipNumber	String	The membership number created for the loyalty program member record.	55.0
transactionJournals	<a href="#">ConnectApi.TransactionJournalOutput</a>	The list of the transaction journal records created after the request is successful.	55.0

## ConnectApi.IndividualMemberOutput

Output representation of the individual member create request.

Property Name	Type	Description	Available Version
contactId	String	The ID of the loyalty program member record created for the individual member.	55.0
loyaltyProgramMemberId	String	The ID of the contact record associated with the loyalty program member record.	55.0
loyaltyProgramName	String	The name of the loyalty program in which the individual member is enrolled.	55.0
membershipNumber	String	The membership number created for the loyalty program member record.	55.0

Property Name	Type	Description	Available Version
transactionJournals	<a href="#">ConnectApi.TransactionOutput</a>	The list of the transaction journal records created after the request is successful.	55.0

## ConnectApi.MemberAccountOutput

Output representation of the details of a loyalty program member's associated account record.

Property Name	Type	Description	Available Version
accountId	String	The ID of the associated account record.	55.0
name	String	The name of the associated account.	55.0

## ConnectApi.MemberContactOutput

Output representation of the details of a loyalty program member's associated contact record.

Property Name	Type	Description	Available Version
contactId	String	The ID of the associated contact record.	55.0
email	String	The email of the associated contact.	55.0
firstName	String	The first name of the associated contact.	55.0
lastName	String	The last name of the associated contact.	55.0

## ConnectApi.MemberDetailsOutput

Output representation of the details of a loyalty program member.

Property Name	Type	Description	Available Version
additionalLoyaltyProgramMemberFields	Map<String, Object>	The details of the custom fields added in the loyalty program member object.	55.0
associatedAccount	<a href="#">ConnectApi.MemberAccountOutput</a>	The details of the account record associated with the loyalty program member.	55.0
associatedContact	<a href="#">ConnectApi.MemberContactOutput</a>	The details of the contact record associated with the loyalty program member.	55.0
canReceivePartnerPromotions	Boolean	Indicates whether the loyalty program member can receive partner promotions ( <code>true</code> ) or not ( <code>false</code> ).	55.0
canReceivePromotions	Boolean	Indicates whether the loyalty program member can receive promotions <code>true</code> ) or not ( <code>false</code> ).	55.0
enrollmentChannel	String	The channel used by the member to enroll in the loyalty program.	55.0

Property Name	Type	Description	Available Version
enrollmentDate	Datetime	The date on which the member enrolled in the loyalty program.	55.0
groupCreatedByMember	String	The membership number of the loyalty program member who created the group.	55.0
groupName	String	The name of the group to which the member belongs.	55.0
lastActivityDate	Datetime	The date on which the member performed their last activity.	55.0
loyaltyProgramMemberId	String	The ID of the loyalty program member.	55.0
loyaltyProgramName	String	The name of the loyalty program.	55.0
memberCurrencies	<a href="#">List&lt;Currency&gt;</a>	The currencies associated with the loyalty program member.	55.0
memberStatus	String	The status of the loyalty program member.	55.0
memberTiers	<a href="#">List&lt;MemberTier&gt;</a>	The tiers associated with the loyalty program member.	55.0
memberType	String	The type of loyalty program member.	55.0
membershipEndDate	Datetime	The date on which the membership ends.	55.0
membershipLastRenewalDate	Datetime	The date on which the loyalty program member last renewed their membership.	55.0
membershipNumber	String	The membership number of the loyalty program member.	55.0
referredBy	String	The membership number of the loyalty program member who referred the member to the loyalty program.	55.0
relatedCorporateMembershipNumber	String	The membership number of the Corporate type loyalty program member associate with the member.	55.0
transactionJournalStatementFrequency	String	The frequency at which transaction journal statements must be delivered to the member.	55.0
transactionJournalStatementLastGeneratedDate	Datetime	The date on which the member's last transaction journal statement was generated.	55.0
transactionJournalStatementMethod	String	The method used to deliver transaction journal statements to the member.	55.0

## ConnectApi.MemberCurrencyOutput

Output representation of the details of the loyalty program member's associated currencies.

Property Name	Type	Description	Available Version
additionalLoyaltyMember CurrencyFields	Map<String, Object>	The details of the custom fields added in the Loyalty Member Currency object.	55.0
escrowPointsBalance	Double	The escrow points balance of the member for the currency.	55.0
expirablePoints	Double	The number of points that the member hasn't redeemed for the currency.	55.0
lastAccrualProcessed Date	Datetime	The date on which the member's accruals for the currency were last processed.	55.0
lastEscrowProcessed Date	Datetime	The date on which the member's escrow points were last rolled over to their points balance for the currency.	55.0
lastExpirationProcess RunDate	Datetime	The date on which the member's points were last expired for the currency.	55.0
lastPoints AggregationDate	Datetime	The date on which the member's points for the fixed type non-qualifying currency were last aggregated.	55.0
lastPointsResetDate	Datetime	The date on which the member's points for the currency were last reset.	55.0
loyaltyMember CurrencyName	String	The name of the Loyalty Member Currency record.	55.0
loyaltyProgram CurrencyId	String	The ID of the loyalty program currency associated with the member.	55.0
loyaltyProgram CurrencyName	String	The name of the loyalty program currency associated with the member.	55.0
memberCurrencyId	String	The ID of the loyalty member currency record.	55.0
nextQualifying PointsResetDate	Datetime	The date on which the member's points balance for the qualifying currency is reset next.	55.0
pointsBalance	Double	The points balance of the member for the currency.	55.0
qualifyingPoints BalanceBeforeReset	Double	The last point balance of the member before the member's points balance was reset to zero for the qualifying currency.	55.0
totalEscrow PointsAccrued	Double	The total number of points that the member has accrued in escrow for the currency.	55.0
totalEscrow RolloverPoints	Double	The total number of points that has rolled over from the member's escrow point balance to the member's points balance for the currency.	55.0



Property Name	Type	Description	Available Version
totalPointsAccrued	Double	The total number of points the member has accrued for the currency.	55.0
totalPointsExpired	Double	The total number of points that have expired for the member.	55.0
totalPointsRedeemed	Double	The total number of points the member has redeemed for the currency.	55.0

## ConnectApi.MemberTierOutput

Output representation of the details of the loyalty program member's associated tiers.

Property Name	Type	Description	Available Version
additionalLoyaltyMemberTierFields	Map<String, Object>	The details of the custom fields added in the Loyalty Member Tier object.	55.0
areTierBenefitsAssigned	Boolean	Indicates whether the benefits associated with the tier have been assigned to the member ( <code>true</code> ) or not ( <code>false</code> ).	55.0
loyaltyMemberTierId	String	The tier ID associated with the loyalty program member.	55.0
loyaltyMemberTierName	String	The name of the loyalty member tier.	55.0
tierChangeReason	String	The reason why the member's was assigned the current tier.	55.0
tierChangeReasonType	String	The reason type why the member's was assigned the current tier.	55.0
tierEffectiveDate	Datetime	The date on which the member was assigned the tier.	55.0
tierExpirationDate	Datetime	The date on which the member's current tier expires.	55.0
tierGroupId	String	The ID of the tier group associated with the Loyalty Tier record.	55.0
tierGroupName	String	The name of the tier group.	55.0
tierId	String	The ID of the loyalty tier record.	55.0
tierSequenceNumber	Integer	The sequence number of the tier.	55.0

## ConnectApi.TransactionJournalOutput

Output representation of the transaction journal records.

Property Name	Type	Description	Available Version
activityDate	Datetime	The date and time of when the transaction journal record is created.	55.0
journalSubType	String	The journal subtype of the transaction journal record.	55.0
journalType	String	The journal type of the transaction journal record.	55.0
loyaltyProgram	String	The loyalty program associated with the transaction journal record.	55.0
loyaltyProgramMember	String	The membership number of the loyalty program member associated with the transaction journal record.	55.0
referredMember	String	The membership number of the loyalty program member who has referred the member.	55.0
status	ConnectApi. JournalStatusResource	The status of the transaction journal record. Possible values are: <ul style="list-style-type: none"> <li>Cancelled</li> <li>Error</li> <li>Pending</li> <li>Processed</li> </ul>	55.0
transactionJournalId	String	The ID of the transaction journal record created after the request is successful.	55.0

## 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. Clients should handle values they don't understand gracefully.

Enum	Description
ConnectApi. EnrollmentChannelResource	Type of loyalty member enrollment channel. <ul style="list-style-type: none"> <li>CallCenter</li> <li>Email</li> <li>Franchise</li> <li>Mobile</li> <li>Partner</li> <li>Pos</li> <li>Print</li> <li>Social</li> <li>Store</li> </ul>

Enum	Description
	<ul style="list-style-type: none"> <li>• Web</li> </ul>
ConnectApi. JournalStatusResource	Type of journal transaction status. <ul style="list-style-type: none"> <li>• Cancelled</li> <li>• Error</li> <li>• Pending</li> <li>• Processed</li> </ul>
ConnectApi. StatementFrequencyResource	Type of statement generation frequency. <ul style="list-style-type: none"> <li>• Monthly</li> <li>• Quaterly</li> </ul>
ConnectApi. StatementMethodResource	Type of statement delivery method. <ul style="list-style-type: none"> <li>• Email</li> <li>• Mail</li> </ul>

## LoyaltyManagement Namespace

The `LoyaltyManagement` namespace provides Apex classes and methods to run the Loyalty Management actions. Use the developer console to invoke these classes and methods. For more information, see [Open the Developer Console](#).

 **Tip:** You can also choose to use the classes and methods of the `Invocable` namespace to call the Loyalty Management invocable actions from Apex.

The following are the classes in the `LoyaltyManagement` namespace.

### [CdpBasedLoyaltyPromotionInput Class](#)

Use this class to retrieve promotions that are eligible for a member based on the member's market segment. This class receives the input parameter values for the Data Cloud based Loyalty Promotion action.

### [CdpBasedLoyaltyPromotionInputBuilder Class](#)

Contains methods to build an instance of the `LoyaltyManagement.CdpBasedLoyaltyPromotionInput` class.

### [CdpBasedLoyaltyPromotionOutput Class](#)

Represents the promotion records that the member is eligible for based on the member's Data Cloud market segment.

### [ChangeTierInput Class](#)

Use this class to change the tier for a given loyalty program member. This class receives the input parameter values for the Change Tier action.

### [ChangeTierInputBuilder Class](#)

Contains methods to build an instance of the `LoyaltyManagement.ChangeTierInput` class.

### [ChangeTierOutput Class](#)

Represents the changed tier details for a loyalty program member.

[CreditPointsOutput Class](#)

Represents the details of points credited to a loyalty program member's balance.

[DebitPointsOutput Class](#)

Represents the details of points debited from a loyalty program member's balance.

[GetMemberActiveSegmentsInput Class](#)

Use this class to retrieve active Data Cloud market segments that a loyalty program member is part of. This class receives the input parameter values for the Get Member's Active Segments action.

[GetMemberActiveSegmentsInputBuilder Class](#)

Contains methods to build an instance of the `LoyaltyManagement.GetMemberActiveSegmentsInput` class.

[GetMemberActiveSegmentsOutput Class](#)

Represents the list of the active segment IDs that the specified loyalty program member is a part of.

[IssueVoucherInput Class](#)

Use this class to issue a voucher to a loyalty program member. This class receives the input parameter values for the Issue Voucher action.

[IssueVoucherInputBuilder Class](#)

Contains methods to build an instance of the `LoyaltyManagement.IssueVoucherInput` class.

[IssueVoucherOutput Class](#)

Represents the voucher ID issued to a loyalty program member.

[LoyaltyActionError Class](#)

A class that contains the errors related to the execution of a Loyalty Management action.

[LoyaltyActionResult Class](#)

A class that contains the result returned by a loyalty action.

[LoyaltyPromotionInput Class](#)

Use this class to retrieve active promotions that are eligible for a transaction journal. The action can process accrual or redemption transactions. This class receives the input parameter values for the Loyalty Promotion action.

[LoyaltyPromotionInputBuilder Class](#)

Contains methods to build an instance of the `LoyaltyManagement.LoyaltyPromotionInput` class.

[LoyaltyPromotionOutput Class](#)

Represents the active promotions based on the specified transaction journal.

[LoyaltyResources Class](#)

A class that executes a loyalty action and returns a result. Takes the input classes as input and returns the objects of output classes.

[MemberPointBalanceInput Class](#)

Use this class to retrieve loyalty points balance for a given loyalty program member. This class receives the input parameter values for the Get Member Points action.

[MemberPointBalanceInputBuilder Class](#)

Contains methods to build an instance of the `LoyaltyManagement.MemberPointsBalanceInput` class.

[MemberPointBalanceOutput Class](#)

Represents the loyalty points balance for a specified loyalty program member.

[MemberTierInput Class](#)

Use this class to retrieve the current tier for a specified loyalty program member. This class receives the input parameter values for the Get Member Tier action.

[MemberTierInputBuilder Class](#)

Contains methods to build an instance of the `LoyaltyManagement.MemberTierInput` class.

[MemberTierOutput Class](#)

Represents the tier details for a specified loyalty program member.

[PointsInput Class](#)

Use this class to credit loyalty points to a loyalty program member's balance or debit loyalty points from a loyalty program member's balance. This class receives the input parameter values for the Credit and Debit point actions.

[PointsInputBuilder Class](#)

Contains methods to build an instance of the `LoyaltyManagement.PointsInput` class.

[TransferMemberPointsToGroupsInput Class](#)

Use this class to transfer points from an individual member or a corporate member to the member's associated group. This class receives the input parameter values for the Transfer Member Points To Groups action.

[TransferMemberPointsToGroupsInputBuilder Class](#)

Contains methods to build an instance of the `LoyaltyManagement.TransferMemberPointsToGroupsInput` class.

[UpdateCumulativeUsageCompletedInput Class](#)

Use this class to update the progress made by a loyalty program member towards attaining a Cumulative type promotion. The action creates a Loyalty Program Member Promotion record if the promotion isn't associated with the member. This class receives the input parameter values for the Update Cumulative Usage Completed action.

[UpdateCumulativeUsageCompletedOutput Class](#)

Represents the details of the progress made by a loyalty program member towards attaining a Cumulative type promotion.

## CdpBasedLoyaltyPromotionInput Class

Use this class to retrieve promotions that are eligible for a member based on the member's market segment. This class receives the input parameter values for the Data Cloud based Loyalty Promotion action.

### Namespace

[LoyaltyManagement](#)

### Example

```
public static void getCdpBasedLoyaltyPromotionsTest() {
    //...Create a list with LoyaltyManagement.CdpBasedLoyaltyPromotionInput as a datatype
    List<LoyaltyManagement.CdpBasedLoyaltyPromotionInput> loyaltyPromotionInputList = new
    List<LoyaltyManagement.CdpBasedLoyaltyPromotionInput>();

    //...Create an instance of the list object and set the input values
    loyaltyPromotionInputList.add(new LoyaltyManagement.CdpBasedLoyaltyPromotionInput input
    = new LoyaltyManagement.CdpBasedLoyaltyPromotionInputBuilder()
        .setMemberId('01MRM000002BD12AM')
        .setPromotions(new List<sObject>{'0c8R000000002MqIAI', '0c8R000000002PqIAI'})
        .build());

    //...Pass the list that contains the input values to the
```

```

LoyaltyResources.getLoyaltyPromotionBasedOnSalesforceCDP() function.
    //...Store the output returned by the function in the CdpBasedLoyaltyPromotionOutput
    object
    List<LoyaltyManagement.CdpBasedLoyaltyPromotionOutput> results =
    LoyaltyManagement.LoyaltyResources.getLoyaltyPromotionBasedOnSalesforceCDP(loyaltyPromotionInputList);

    System.debug('Get Cdp Based Promotions:'+ results.get(0).isSuccess());
}

```

## CdpBasedLoyaltyPromotionInputBuilder Class

Contains methods to build an instance of the `LoyaltyManagement.CdpBasedLoyaltyPromotionInput` class.

### Namespace

[LoyaltyManagement](#)

[CdpBasedLoyaltyPromotionInputBuilder Methods](#)

### CdpBasedLoyaltyPromotionInputBuilder Methods

The following are methods for `CdpBasedLoyaltyPromotionInputBuilder`.

#### [build\(\)](#)

Returns an instance of the `LoyaltyManagement.CdpBasedLoyaltyPromotionInput` object.

#### [setMemberId\(memberId\)](#)

Represents the ID of a loyalty program member for whom promotions must be filtered. This is a required field. Ensure that the member is part of a Data Cloud market segment.

#### [setPromotions\(promotions\)](#)

Represents the promotion records to filter the promotions that are applicable to the member's market segment.

### **build()**

Returns an instance of the `LoyaltyManagement.CdpBasedLoyaltyPromotionInput` object.

### Signature

```
public LoyaltyManagement.CdpBasedLoyaltyPromotionInput build()
```

### Return Value

Type: [LoyaltyManagement.CdpBasedLoyaltyPromotionInput](#)

### **setMemberId (memberId)**

Represents the ID of a loyalty program member for whom promotions must be filtered. This is a required field. Ensure that the member is part of a Data Cloud market segment.

### Signature

```
public LoyaltyManagement.CdpBasedLoyaltyPromotionInputBuilder setMemberId(String  
memberId)
```

### Parameters

*memberId*  
Type: String

### Return Value

Type: [LoyaltyManagement.CdpBasedLoyaltyPromotionInputBuilder](#)

### **setPromotions (promotions)**

Represents the promotion records to filter the promotions that are applicable to the member's market segment.

### Signature

```
public LoyaltyManagement.CdpBasedLoyaltyPromotionInputBuilder setPromotions(List<SObject>  
promotions)
```

### Parameters

*promotions*  
Type: List<SObject>

### Return Value

Type: [LoyaltyManagement.CdpBasedLoyaltyPromotionInputBuilder](#)

## CdpBasedLoyaltyPromotionOutput Class

Represents the promotion records that the member is eligible for based on the member's Data Cloud market segment.

## Namespace

[LoyaltyManagement](#)

[CdpBasedLoyaltyPromotionOutput Methods](#)

## CdpBasedLoyaltyPromotionOutput Methods

The following are methods for [CdpBasedLoyaltyPromotionOutput](#).

[getErrors\(\)](#)

If an error occurred, returns a list providing the error code and description.

**getPromotions()**

Returns a List of sObject data type that contains the promotion records.

**isSuccess()**

Returns a Boolean that is set to true if the Data Cloud based loyalty promotion operation was successful for this object, false otherwise.

**getErrors ()**

If an error occurred, returns a list providing the error code and description.

**Signature**

```
public List<LoyaltyManagement.LoyaltyActionError> getErrors ()
```

**Return Value**

Type: List<LoyaltyManagement.LoyaltyActionError>

**getPromotions ()**

Returns a List of sObject data type that contains the promotion records.

**Signature**

```
public List<SObject> getPromotions ()
```

**Return Value**

Type: List<SObject>

**isSuccess ()**

Returns a Boolean that is set to true if the Data Cloud based loyalty promotion operation was successful for this object, false otherwise.

**Signature**

```
public Boolean isSuccess ()
```

**Return Value**

Type: Boolean

## ChangeTierInput Class

Use this class to change the tier for a given loyalty program member. This class receives the input parameter values for the Change Tier action.

**Namespace**

[LoyaltyManagement](#)



## Example

```
public void changeTierTest(){
    //...Create a list with LoyaltyManagement.ChangeTierInput as a datatype
    List<LoyaltyManagement.ChangeTierInput> changeTierList = new
    List<LoyaltyManagement.ChangeTierInput>();

    //...Create an instance of the list object and set the input values
    changeTierList.add(new LoyaltyManagement.ChangeTierInputBuilder()
        .setLoyaltyProgramMemberId('01cxx00006000001AAA')
        .setProgramName('baProgram')
        .setTierGroupName('QualifyingGroup')
        .setTargetTierName('Gold')
        .setChangeReason('Reason for change.')
        .build());

    //...Pass the list that contains the input values to the LoyaltyResources.changeTier()
    function.
    //...Store the output returned by the function in the ChangeTierOutput object
    List<LoyaltyManagement.ChangeTierOutput> changeTierActionResults =
    LoyaltyManagement.LoyaltyResources.changeTier(changeTierList);
    System.debug('Change Tier Successful Operation:'+
    changeTierActionResults.get(0).isSuccess());
}
```

## ChangeTierInputBuilder Class

Contains methods to build an instance of the `LoyaltyManagement.ChangeTierInput` class.

### Namespace

[LoyaltyManagement](#)

[ChangeTierInputBuilder Methods](#)

### ChangeTierInputBuilder Methods

The following are methods for `ChangeTierInputBuilder`.

[build\(\)](#)

Returns an instance of the `LoyaltyManagement.ChangeTierInput` object.

[setChangeReason\(changeReason\)](#)

Represents the reason to change a member tier.

[setLoyaltyProgramMemberId\(loyaltyProgramMemberId\)](#)

Represents the ID of the loyalty program member. This is a required field.

[setProgramName\(programName\)](#)

Represents the loyalty program name associated with the tier. This is a required field.

[setTargetTierName\(targetTierName\)](#)

Represents the new tier to associate with the loyalty program member. This is a required field.

[setTierGroupName\(tierGroupName\)](#)

Represents the name of the loyalty tier group. This is a required field.

### **build()**

Returns an instance of the `LoyaltyManagement.ChangeTierInput` object.

### Signature

```
public LoyaltyManagement.ChangeTierInput build()
```

### Return Value

Type: [LoyaltyManagement.ChangeTierInput](#)

### **setChangeReason (changeReason)**

Represents the reason to change a member tier.

### Signature

```
public LoyaltyManagement.ChangeTierInputBuilder setChangeReason(String changeReason)
```

### Parameters

*changeReason*

Type: String

### Return Value

Type: [LoyaltyManagement.ChangeTierInputBuilder](#)

### **setLoyaltyProgramMemberId (loyaltyProgramMemberId)**

Represents the ID of the loyalty program member. This is a required field.

### Signature

```
public LoyaltyManagement.ChangeTierInputBuilder setLoyaltyProgramMemberId(String loyaltyProgramMemberId)
```

### Parameters

*loyaltyProgramMemberId*

Type: String

## Return Value

Type: [LoyaltyManagement.ChangeTierInputBuilder](#)

### **setProgramName (programName)**

Represents the loyalty program name associated with the tier. This is a required field.

## Signature

```
public LoyaltyManagement.ChangeTierInputBuilder setProgramName (String programName)
```

## Parameters

*programName*  
Type: String

## Return Value

Type: [LoyaltyManagement.ChangeTierInputBuilder](#)

### **setTargetTierName (targetTierName)**

Represents the new tier to associate with the loyalty program member. This is a required field.

## Signature

```
public LoyaltyManagement.ChangeTierInputBuilder setTargetTierName (String targetTierName)
```

## Parameters

*targetTierName*  
Type: String

## Return Value

Type: [LoyaltyManagement.ChangeTierInputBuilder](#)

### **setTierGroupName (tierGroupName)**

Represents the name of the loyalty tier group. This is a required field.

## Signature

```
public LoyaltyManagement.ChangeTierInputBuilder setTierGroupName (String tierGroupName)
```

## Parameters

*tierGroupName*  
Type: String

## Return Value

Type: [LoyaltyManagement.ChangeTierInputBuilder](#)

# ChangeTierOutput Class

Represents the changed tier details for a loyalty program member.

## Namespace

[LoyaltyManagement](#)

[ChangeTierOutput Methods](#)

## ChangeTierOutput Methods

The following are methods for `ChangeTierOutput`.

[getErrors\(\)](#)

If an error occurred, returns a list providing the error code and description.

[getLoyaltyProgramMemberId\(\)](#)

Returns a String that contains the loyalty program member ID.

[getMemberTier\(\)](#)

Returns a String that contains the tier associated with the loyalty program member.

[getProgramName\(\)](#)

Returns a String that contains the loyalty program name.

[getTierGroupName\(\)](#)

Returns a String that contains the loyalty tier group name.

[isSuccess\(\)](#)

Returns a Boolean that is set to true if the change tier operation was successful for this object, false otherwise.

### **getErrors ()**

If an error occurred, returns a list providing the error code and description.

## Signature

```
public List<LoyaltyManagement.LoyaltyActionError> getErrors ()
```

## Return Value

Type: `List<LoyaltyManagement.LoyaltyActionError>`

### **getLoyaltyProgramMemberId ()**

Returns a String that contains the loyalty program member ID.

**Signature**

```
public String getLoyaltyProgramMemberId()
```

**Return Value**

Type: String

**getMemberTier()**

Returns a String that contains the tier associated with the loyalty program member.

**Signature**

```
public String getMemberTier()
```

**Return Value**

Type: String

**getProgramName()**

Returns a String that contains the loyalty program name.

**Signature**

```
public String getProgramName()
```

**Return Value**

Type: String

**getTierGroupName()**

Returns a String that contains the loyalty tier group name.

**Signature**

```
public String getTierGroupName()
```

**Return Value**

Type: String

**isSuccess()**

Returns a Boolean that is set to true if the change tier operation was successful for this object, false otherwise.

**Signature**

```
public Boolean isSuccess()
```

## Return Value

Type: Boolean

# CreditPointsOutput Class

Represents the details of points credited to a loyalty program member's balance.

## Namespace

[LoyaltyManagement](#)

[CreditPointsOutput Methods](#)

## CreditPointsOutput Methods

The following are methods for `CreditPointsOutput`.

[getCurrencyName\(\)](#)

Returns a String that contains the name of the program currency.

[getErrors\(\)](#)

If an error occurred, returns a list providing the error code and description.

[getJournalId\(\)](#)

Returns a String that contains the ID of the transaction journal.

[getLoyaltyProgramMemberId\(\)](#)

Returns a String that contains the ID of the loyalty program member.

[getProgramName\(\)](#)

Returns a String that contains the name of the loyalty program.

[isSuccess\(\)](#)

Returns a Boolean that is set to true if the credit points operation was successful for this object, false otherwise.

### **getCurrencyName ()**

Returns a String that contains the name of the program currency.

## Signature

```
public String getCurrencyName ()
```

## Return Value

Type: String

### **getErrors ()**

If an error occurred, returns a list providing the error code and description.

### Signature

```
public List<LoyaltyManagement.LoyaltyActionError> getErrors()
```

### Return Value

Type: List<LoyaltyManagement.LoyaltyActionError>

### **getJournalId()**

Returns a String that contains the ID of the transaction journal.

### Signature

```
public String getJournalId()
```

### Return Value

Type: String

### **getLoyaltyProgramMemberId()**

Returns a String that contains the ID of the loyalty program member.

### Signature

```
public String getLoyaltyProgramMemberId()
```

### Return Value

Type: String

### **getProgramName()**

Returns a String that contains the name of the loyalty program.

### Signature

```
public String getProgramName()
```

### Return Value

Type: String

### **isSuccess()**

Returns a Boolean that is set to true if the credit points operation was successful for this object, false otherwise.

### Signature

```
public Boolean isSuccess()
```

## Return Value

Type: Boolean

# DebitPointsOutput Class

Represents the details of points debited from a loyalty program member's balance.

## Namespace

[LoyaltyManagement](#)

[DebitPointsOutput Methods](#)

## DebitPointsOutput Methods

The following are methods for `DebitPointsOutput`.

[getCurrencyName\(\)](#)

Returns a String that contains the name of the program currency.

[getErrors\(\)](#)

If an error occurred, returns a list providing the error code and description.

[getJournalId\(\)](#)

Returns a String that contains the ID of the transaction journal.

[getLoyaltyProgramMemberId\(\)](#)

Returns a String that contains the ID of the loyalty program member.

[getPoints\(\)](#)

Returns a Double that contains the total points available for the loyalty program member.

[getProgramName\(\)](#)

Returns a String that contains the name of the loyalty program.

[isSuccess\(\)](#)

Returns a Boolean that is set to true if the debit points operation was successful for this object, false otherwise.

### **getCurrencyName ()**

Returns a String that contains the name of the program currency.

## Signature

```
public String getCurrencyName ()
```

## Return Value

Type: String



**getErrors ()**

If an error occurred, returns a list providing the error code and description.

**Signature**

```
public List<LoyaltyManagement.LoyaltyActionError> getErrors ()
```

**Return Value**

Type: List<LoyaltyManagement.LoyaltyActionError>

**getJournalId ()**

Returns a String that contains the ID of the transaction journal.

**Signature**

```
public String getJournalId ()
```

**Return Value**

Type: String

**getLoyaltyProgramMemberId ()**

Returns a String that contains the ID of the loyalty program member.

**Signature**

```
public String getLoyaltyProgramMemberId ()
```

**Return Value**

Type: String

**getPoints ()**

Returns a Double that contains the total points available for the loyalty program member.

**Signature**

```
public Double getPoints ()
```

**Return Value**

Type: Double

**getProgramName ()**

Returns a String that contains the name of the loyalty program.

### Signature

```
public String getProgramName()
```

### Return Value

Type: String

### isSuccess ()

Returns a Boolean that is set to true if the debit points operation was successful for this object, false otherwise.

### Signature

```
public Boolean isSuccess()
```

### Return Value

Type: Boolean

## GetMemberActiveSegmentsInput Class

Use this class to retrieve active Data Cloud market segments that a loyalty program member is part of. This class receives the input parameter values for the Get Member's Active Segments action.

### Namespace

[LoyaltyManagement](#)

## GetMemberActiveSegmentsInputBuilder Class

Contains methods to build an instance of the `LoyaltyManagement.GetMemberActiveSegmentsInput` class.

### Namespace

[LoyaltyManagement](#)

[GetMemberActiveSegmentsInputBuilder Methods](#)

## GetMemberActiveSegmentsInputBuilder Methods

The following are methods for `GetMemberActiveSegmentsInputBuilder`.

[build\(\)](#)

Returns an instance of the `LoyaltyManagement.GetMemberActiveSegmentsInput` object.

[setLoyaltyProgramMemberID\(loyaltyProgramMemberID\)](#)

Represents the ID of the loyalty program member to retrieve associated active segments for. This is a required field.

[setSegmentIds\(segmentIds\)](#)

Represents a comma-delimited list of segmentIds to retrieve for the specified member's associated segments.

**build()**

Returns an instance of the `LoyaltyManagement.GetMemberActiveSegmentsInput` object.

**Signature**

```
public LoyaltyManagement.GetMemberActiveSegmentsInput build()
```

**Return Value**

Type: [LoyaltyManagement.GetMemberActiveSegmentsInput](#)

**setLoyaltyProgramMemberID(loyaltyProgramMemberID)**

Represents the ID of the loyalty program member to retrieve associated active segments for. This is a required field.

**Signature**

```
public LoyaltyManagement.GetMemberActiveSegmentsInputBuilder  
setLoyaltyProgramMemberID(String loyaltyProgramMemberID)
```

**Parameters**

*loyaltyProgramMemberID*

Type: String

**Return Value**

Type: [LoyaltyManagement.GetMemberActiveSegmentsInputBuilder](#)

**setSegmentIds(segmentIds)**

Represents a comma-delimited list of segmentIds to retrieve for the specified member's associated segments.

**Signature**

```
public LoyaltyManagement.GetMemberActiveSegmentsInputBuilder  
setSegmentIds(List<String>  
segmentIds)
```

**Parameters**

*segmentIds*

Type: List<String>

**Return Value**

Type: [LoyaltyManagement.GetMemberActiveSegmentsInputBuilder](#)

## GetMemberActiveSegmentsOutput Class

Represents the list of the active segment IDs that the specified loyalty program member is a part of.

### Namespace

[LoyaltyManagement](#)

[GetMemberActiveSegmentsOutput Methods](#)

### GetMemberActiveSegmentsOutput Methods

The following are methods for `GetMemberActiveSegmentsOutput`.

[getErrors\(\)](#)

If an error occurred, returns a list providing the error code and description.

[getMemberSegmentIDs\(\)](#)

Returns a comma-delimited list of the active segment IDs that the specified loyalty program member is a part of.

[isSuccess\(\)](#)

Returns a Boolean that is set to true if the get member active segments operation was successful for this object, false otherwise.

#### **getErrors ()**

If an error occurred, returns a list providing the error code and description.

#### Signature

```
public List<LoyaltyManagement.LoyaltyActionError> getErrors ()
```

#### Return Value

Type: `List<LoyaltyManagement.LoyaltyActionError>`

#### **getMemberSegmentIDs ()**

Returns a comma-delimited list of the active segment IDs that the specified loyalty program member is a part of.



**Note:** If input segment IDs are passed, then the response is a subset of the specified segment IDs.

#### Signature

```
public List<String> getMemberSegmentIDs ()
```

#### Return Value

Type: `List<String>`

### **isSuccess ()**

Returns a Boolean that is set to true if the get member active segments operation was successful for this object, false otherwise.

### Signature

```
public Boolean isSuccess ()
```

### Return Value

Type: Boolean

## IssueVoucherInput Class

Use this class to issue a voucher to a loyalty program member. This class receives the input parameter values for the Issue Voucher action.

## Namespace

[LoyaltyManagement](#)

## Example

```
public void issueVoucherTest () {
    //...Create a list with LoyaltyManagement.IssueVoucherInput as a datatype
    List<LoyaltyManagement.IssueVoucherInput> issueVoucherList = new
    List<LoyaltyManagement.IssueVoucherInput> ();

    //...Create an instance of the list object and set the input values
    issueVoucherList.add(new
    LoyaltyManagement.IssueVoucherInputBuilder ().setLoyaltyProgramMemberId('01MRM0000002BD12AM')

        .setProgramName('LP01')
        .setVoucherDefinitionName('V1')
        .setVoucherCode('demoCode')
        .setJournalId('01VRM0000004DH22AM')
        .build());

    //...Pass the list that contains the input values to the LoyaltyResources.issueVoucher()
    function.
    //...Store the output returned by the function in the IssueVoucherOutput object
    List<LoyaltyManagement.IssueVoucherOutput> issueVoucherActionResults =
    LoyaltyManagement.LoyaltyResources.issueVoucher(issueVoucherList);
    System.debug('Issue Voucher Successful Operation:'+ issueVoucherActionResults.get(0));
}
```

## IssueVoucherInputBuilder Class

Contains methods to build an instance of the `LoyaltyManagement.IssueVoucherInput` class.

## Namespace

[LoyaltyManagement](#)

[IssueVoucherInputBuilder Methods](#)

## IssueVoucherInputBuilder Methods

The following are methods for `IssueVoucherInputBuilder`.

[build\(\)](#)

Returns an instance of the `LoyaltyManagement.IssueVoucherInput` object.

[setAdditionalNotes\(additionalNotes\)](#)

Represents extra information about the voucher.

[setBenefitId\(benefitId\)](#)

Represents the benefit associated with the voucher.

[setContactId\(contactId\)](#)

Represents the contact associated with the loyalty program member.

[setDiscountPercent\(discountPercent\)](#)

Represents the discount percentage provided by the voucher.

[setEffectiveDate\(effectiveDate\)](#)

Represents the start date of the validity of the voucher.

[setExpiryDate\(expiryDate\)](#)

Represents the expiry date of the voucher.

[setFaceValue\(faceValue\)](#)

Represents the amount of discount provided by the voucher.

[setJournalId\(journalId\)](#)

Represents the transaction journal associated with the member voucher when the voucher results from accrual of points.

[setLoyaltyProgramMemberId\(loyaltyProgramMemberId\)](#)

Represents the ID of the loyalty program member to issue the voucher.

[setProgramName\(programName\)](#)

Represents the name of the loyalty program associated with the voucher.

[setPromotionId\(promotionId\)](#)

Represents the promotion associated with the voucher.

[setVoucherCode\(voucherCode\)](#)

Represents the voucher identifier generated by an external system. This is a required field.

[setVoucherDefinitionName\(voucherDefinitionName\)](#)

Represents the name of the voucher definition associated with the voucher code. This is a required field.

### **build()**

Returns an instance of the `LoyaltyManagement.IssueVoucherInput` object.

### Signature

```
public LoyaltyManagement.IssueVoucherInput build()
```

### Return Value

Type: [LoyaltyManagement.IssueVoucherInput](#)

### **setAdditionalNotes (additionalNotes)**

Represents extra information about the voucher.

### Signature

```
public LoyaltyManagement.IssueVoucherInputBuilder setAdditionalNotes(String additionalNotes)
```

### Parameters

*additionalNotes*  
Type: String

### Return Value

Type: [LoyaltyManagement.IssueVoucherInputBuilder](#)

### **setBenefitId (benefitId)**

Represents the benefit associated with the voucher.

### Signature

```
public LoyaltyManagement.IssueVoucherInputBuilder setBenefitId(String benefitId)
```

### Parameters

*benefitId*  
Type: String

### Return Value

Type: [LoyaltyManagement.IssueVoucherInputBuilder](#)

### **setContactId (contactId)**

Represents the contact associated with the loyalty program member.

### Signature

```
public LoyaltyManagement.IssueVoucherInputBuilder setContactId(String contactId)
```

## Parameters

*contactId*  
Type: String

## Return Value

Type: [LoyaltyManagement.IssueVoucherInputBuilder](#)

### **setDiscountPercent (discountPercent)**

Represents the discount percentage provided by the voucher.

## Signature

```
public LoyaltyManagement.IssueVoucherInputBuilder setDiscountPercent(Integer discountPercent)
```

## Parameters

*discountPercent*  
Type: Integer

The discount percentage provided by the voucher. Specify a value to override the discount percentage specified in the associated voucher definition. Specify this when the associated voucher definition is of the type Discount Percentage. This field is available in API version 53.0 and later.

## Return Value

Type: [LoyaltyManagement.IssueVoucherInputBuilder](#)

### **setEffectiveDate (effectiveDate)**

Represents the start date of the validity of the voucher.

## Signature

```
public LoyaltyManagement.IssueVoucherInputBuilder setEffectiveDate(String effectiveDate)
```

## Parameters

*effectiveDate*  
Type: String

The supported date format is YYYY-MM-DD.

## Return Value

Type: [LoyaltyManagement.IssueVoucherInputBuilder](#)



**setExpiryDate (expiryDate)**

Represents the expiry date of the voucher.

**Signature**

```
public LoyaltyManagement.IssueVoucherInputBuilder setExpiryDate(String expiryDate)
```

**Parameters**

*expiryDate*

Type: String

The supported date format is YYYY-MM-DD.

**Return Value**

Type: [LoyaltyManagement.IssueVoucherInputBuilder](#)

**setFaceValue (faceValue)**

Represents the amount of discount provided by the voucher.

**Signature**

```
public LoyaltyManagement.IssueVoucherInputBuilder setFaceValue(Double faceValue)
```

**Parameters**

*faceValue*

Type: Double

The amount of discount provided by the voucher. Specify a value to override the face value specified in the associated voucher definition. Specify this when the associated voucher definition is of the type Fixed Value. This field is available in API version 53.0 and later.

**Return Value**

Type: [LoyaltyManagement.IssueVoucherInputBuilder](#)

**setJournalId (journalId)**

Represents the transaction journal associated with the member voucher when the voucher results from accrual of points.

**Signature**

```
public LoyaltyManagement.IssueVoucherInputBuilder setJournalId(String journalId)
```

**Parameters**

*journalId*

Type: String

## Return Value

Type: [LoyaltyManagement.IssueVoucherInputBuilder](#)

### **setLoyaltyProgramMemberId (loyaltyProgramMemberId)**

Represents the ID of the loyalty program member to issue the voucher.

## Signature

```
public LoyaltyManagement.IssueVoucherInputBuilder setLoyaltyProgramMemberId(String  
loyaltyProgramMemberId)
```

## Parameters

*loyaltyProgramMemberId*  
Type: String

## Return Value

Type: [LoyaltyManagement.IssueVoucherInputBuilder](#)

### **setProgramName (programName)**

Represents the name of the loyalty program associated with the voucher.

## Signature

```
public LoyaltyManagement.IssueVoucherInputBuilder setProgramName(String programName)
```

## Parameters

*programName*  
Type: String

## Return Value

Type: [LoyaltyManagement.IssueVoucherInputBuilder](#)

### **setPromotionId (promotionId)**

Represents the promotion associated with the voucher.

## Signature

```
public LoyaltyManagement.IssueVoucherInputBuilder setPromotionId(String promotionId)
```

## Parameters

*promotionId*  
Type: String

## Return Value

Type: [LoyaltyManagement.IssueVoucherInputBuilder](#)

### **setVoucherCode (voucherCode)**

Represents the voucher identifier generated by an external system. This is a required field.

## Signature

```
public LoyaltyManagement.IssueVoucherInputBuilder setVoucherCode(String voucherCode)
```

## Parameters

*voucherCode*

Type: String

## Return Value

Type: [LoyaltyManagement.IssueVoucherInputBuilder](#)

### **setVoucherDefinitionName (voucherDefinitionName)**

Represents the name of the voucher definition associated with the voucher code. This is a required field.

## Signature

```
public LoyaltyManagement.IssueVoucherInputBuilder setVoucherDefinitionName(String voucherDefinitionName)
```

## Parameters

*voucherDefinitionName*

Type: String

## Return Value

Type: [LoyaltyManagement.IssueVoucherInputBuilder](#)

## IssueVoucherOutput Class

Represents the voucher ID issued to a loyalty program member.

## Namespace

[LoyaltyManagement](#)

[IssueVoucherOutput Methods](#)

## IssueVoucherOutput Methods

The following are methods for `IssueVoucherOutput`.

### [getErrors\(\)](#)

If an error occurred, returns a list providing the error code and description.

### [getVoucherId\(\)](#)

Returns the ID of the voucher that's generated after the request is successful.

### [isSuccess\(\)](#)

Returns a Boolean that is set to true if the issue voucher operation was successful for this object, false otherwise.

### **getErrors ()**

If an error occurred, returns a list providing the error code and description.

#### Signature

```
public List<LoyaltyManagement.LoyaltyActionError> getErrors ()
```

#### Return Value

Type: `List<LoyaltyManagement.LoyaltyActionError>`

### **getVoucherId ()**

Returns the ID of the voucher that's generated after the request is successful.

#### Signature

```
public String getVoucherId ()
```

#### Return Value

Type: `String`

### **isSuccess ()**

Returns a Boolean that is set to true if the issue voucher operation was successful for this object, false otherwise.

#### Signature

```
public Boolean isSuccess ()
```

#### Return Value

Type: `Boolean`

## LoyaltyActionError Class

A class that contains the errors related to the execution of a Loyalty Management action.

### Namespace

[LoyaltyManagement](#)

### Example

```
public void creditPointsTestError() {
    List<LoyaltyManagement.PointsInput> creditPointsList = new
List<LoyaltyManagement.PointsInput> ();
    creditPointsList.add(new LoyaltyManagement.PointsInputBuilder ()
        .setLoyaltyProgramMemberId('01MRM0000002BD12AM')
        .setProgramName('LP01')
        .setProgramCurrencyName('Miles')
        .setPoints(1200)
        .setJournalId('01VT10000008Cn1MAE')
        .build());
    List<LoyaltyManagement.CreditPointsOutput> creditPointActionResults =
LoyaltyManagement.LoyaltyResources.creditPoints(creditPointsList);
    System.debug('Is Success:' + creditPointActionResults.get(0).isSuccess());
    System.debug('Errors:' + creditPointActionResults.get(0).getErrors());
}
```

[LoyaltyActionError Constructors](#)

[LoyaltyActionError Methods](#)

### LoyaltyActionError Constructors

The following are constructors for `LoyaltyActionError`.

[LoyaltyActionError\(code, message\)](#)

Creates an instance of the `LoyaltyManagement.LoyaltyActionError` class.

#### **LoyaltyActionError (code, message)**

Creates an instance of the `LoyaltyManagement.LoyaltyActionError` class.

### Signature

```
public LoyaltyActionError(String code, String message)
```

### Parameters

*code*

Type: String

The error code.

*message*

Type: String

The error message.

## LoyaltyActionError Methods

The following are methods for `LoyaltyActionError`.

[getCode\(\)](#)

Returns a String that contains an error code.

[getMessage\(\)](#)

Returns a String that contains an error description.

### **getCode ()**

Returns a String that contains an error code.

### Signature

```
public String getCode ()
```

### Return Value

Type: String

### **getMessage ()**

Returns a String that contains an error description.

### Signature

```
public String getMessage ()
```

### Return Value

Type: String

## LoyaltyActionResult Class

A class that contains the result returned by a loyalty action.

## Namespace

[LoyaltyManagement](#)

[LoyaltyActionResult Methods](#)

## LoyaltyActionResult Methods

The following are methods for `LoyaltyActionResult`.

### `getErrors()`

If an error occurred, returns a `LoyaltyManagement.LoyaltyActionError` object providing the error code and description.

### `isSuccess()`

Returns a Boolean that is set to true if the Loyalty Management action operation was successful for this object, false otherwise.

### **`getErrors()`**

If an error occurred, returns a `LoyaltyManagement.LoyaltyActionError` object providing the error code and description.

### Signature

```
public List<LoyaltyManagement.LoyaltyActionError> getErrors()
```

### Return Value

Type: `List<LoyaltyManagement.LoyaltyActionError>`

### **`isSuccess()`**

Returns a Boolean that is set to true if the Loyalty Management action operation was successful for this object, false otherwise.

### Signature

```
public Boolean isSuccess()
```

### Return Value

Type: Boolean

## LoyaltyPromotionInput Class

Use this class to retrieve active promotions that are eligible for a transaction journal. The action can process accrual or redemption transactions. This class receives the input parameter values for the Loyalty Promotion action.

## Namespace

[LoyaltyManagement](#)

## Example

```
public static void getLoyaltyPromotionsTest() {
    //...Create a list with 'LoyaltyManagement.LoyaltyPromotionInput' as a datatype
    List<LoyaltyManagement.LoyaltyPromotionInput> loyaltyPromotionInputList = new
    List<LoyaltyManagement.LoyaltyPromotionInput>();
}
```

```
//...Create an instance of the list object and set the input values
loyaltyPromotionInputList.add(new LoyaltyManagement.LoyaltyPromotionInput input = new
LoyaltyManagement.LoyaltyPromotionInputBuilder()
    .setJournalId('01VRM0000004DH22AM')
    .build());

//...Pass the list that contains the input values to the
'LoyaltyResources.getLoyaltyPromotions()' function.
//...Store the output returned by the function in the 'LoyaltyPromotionOutput' object
List<LoyaltyManagement.LoyaltyPromotionOutput> results =
LoyaltyManagement.LoyaltyResources.getLoyaltyPromotions(loyaltyPromotionInputList);
System.debug('Get Loyalty Promotions:'+ results.get(0).isSuccess());
}
```

## LoyaltyPromotionInputBuilder Class

Contains methods to build an instance of the `LoyaltyManagement.LoyaltyPromotionInput` class.

### Namespace

[LoyaltyManagement](#)

[LoyaltyPromotionInputBuilder Methods](#)

### LoyaltyPromotionInputBuilder Methods

The following are methods for `LoyaltyPromotionInputBuilder`.

[build\(\)](#)

Returns an instance of the `LoyaltyManagement.LoyaltyPromotionInput` object.

[setJournalId\(journalId\)](#)

Represents the ID of an Accrual or Redemption type Transaction Journal record that retrieval of active loyalty promotions is based on. This is a required field.

#### **build()**

Returns an instance of the `LoyaltyManagement.LoyaltyPromotionInput` object.

#### Signature

```
public LoyaltyManagement.LoyaltyPromotionInput build()
```

#### Return Value

Type: [LoyaltyManagement.LoyaltyPromotionInput](#)



**setJournalId(journalId)**

Represents the ID of an Accrual or Redemption type Transaction Journal record that retrieval of active loyalty promotions is based on. This is a required field.

**Signature**

```
public LoyaltyManagement.LoyaltyPromotionInputBuilder setJournalId(String journalId)
```

**Parameters**

*journalId*  
Type: String

**Return Value**

Type: [LoyaltyManagement.LoyaltyPromotionInputBuilder](#)

## LoyaltyPromotionOutput Class

Represents the active promotions based on the specified transaction journal.

### Namespace

[LoyaltyManagement](#)

[LoyaltyPromotionOutput Methods](#)

### LoyaltyPromotionOutput Methods

The following are methods for `LoyaltyPromotionOutput`.

[getErrors\(\)](#)

If an error occurred, returns a list providing the error code and description.

[getPromotions\(\)](#)

Returns a List of sObject data type that contains the promotion records that are eligible for the transaction journal.

[isSuccess\(\)](#)

Returns a Boolean that is set to true if the loyalty promotion operation was successful for this object, false otherwise.

**getErrors ()**

If an error occurred, returns a list providing the error code and description.

**Signature**

```
public List<LoyaltyManagement.LoyaltyActionError> getErrors ()
```

### Return Value

Type: List<[LoyaltyManagement.LoyaltyActionError](#)>

### **getPromotions ()**

Returns a List of sObject data type that contains the promotion records that are eligible for the transaction journal.

### Signature

```
public List<SObject> getPromotions ()
```

### Return Value

Type: List<SObject>

### **isSuccess ()**

Returns a Boolean that is set to true if the loyalty promotion operation was successful for this object, false otherwise.

### Signature

```
public Boolean isSuccess ()
```

### Return Value

Type: Boolean

## LoyaltyResources Class

A class that executes a loyalty action and returns a result. Takes the input classes as input and returns the objects of output classes.

## Namespace

[LoyaltyManagement](#)

[LoyaltyResources Methods](#)

## LoyaltyResources Methods

The following are methods for `LoyaltyResources`.

[changeTier\(changeTierList\)](#)

Use this method to change the tier for a loyalty program member.

[creditPoints\(creditPointsList\)](#)

Use this method to credit loyalty points to a loyalty program member's balance.

[debitPoints\(debitPointsList\)](#)

Use this method to debit loyalty points from a loyalty program member's balance.

[getLoyaltyPromotionBasedOnSalesforceCDP\(loyaltyPromotionInputList\)](#)

Use this method to retrieve promotions that are eligible for a member based on the member's market segment.

[getLoyaltyPromotions\(loyaltyPromotionInputList\)](#)

Use this method to retrieve active promotions that are eligible for a transaction journal. The action can process accrual or redemption transactions.

[getPointsBalance\(memberPointBalanceList\)](#)

Use this method to retrieve the loyalty points balance for a loyalty program member.

[getTier\(memberTierList\)](#)

Use this method to retrieve the current tier for a loyalty program member.

[issueVoucher\(issueVoucherList\)](#)

Use this method to issue a voucher for a loyalty program member.

[transferMemberPointsToGroups\(transferMemberPointsToGroupsList\)](#)

Use this method to transfer points from an individual member or a corporate member to the member's associated group.

[updateProgressForCumulativePromotionUsage\(updateCumulativeUsageCompletedInputList\)](#)

Use this method to update the progress made by a loyalty program member towards attaining a Cumulative type promotion.

**changeTier (changeTierList)**

Use this method to change the tier for a loyalty program member.

**Signature**

```
public static List<LoyaltyManagement.ChangeTierOutput>
changeTier (List<LoyaltyManagement.ChangeTierInput> changeTierList)
```

**Parameters**

*changeTierList*

Type: List<LoyaltyManagement.ChangeTierInput>

Set of inputs to change the tier for a loyalty program member.

**Return Value**

Type: List<LoyaltyManagement.ChangeTierOutput>

**creditPoints (creditPointsList)**

Use this method to credit loyalty points to a loyalty program member's balance.

**Signature**

```
public static List<LoyaltyManagement.CreditPointsOutput>
creditPoints (List<LoyaltyManagement.PointsInput> creditPointsList)
```

## Parameters

*creditPointsList*

Type: List<LoyaltyManagement.PointsInput>

Set of inputs to credit points to a loyalty program member's balance.

## Return Value

Type: List<LoyaltyManagement.CreditPointsOutput>

## **debitPoints (debitPointsList)**

Use this method to debit loyalty points from a loyalty program member's balance.

## Signature

```
public static List<LoyaltyManagement.DebitPointsOutput>  
debitPoints (List<LoyaltyManagement.PointsInput> debitPointsList)
```

## Parameters

*debitPointsList*

Type: List<LoyaltyManagement.PointsInput>

Set of inputs to debit points from a loyalty program member's balance.

## Return Value

Type: List<LoyaltyManagement.DebitPointsOutput>

## **getLoyaltyPromotionBasedOnSalesforceCDP (loyaltyPromotionInputList)**

Use this method to retrieve promotions that are eligible for a member based on the member's market segment.

## Signature

```
public static List<LoyaltyManagement.CdpBasedLoyaltyPromotionOutput>  
getLoyaltyPromotionBasedOnSalesforceCDP (List<LoyaltyManagement.CdpBasedLoyaltyPromotionInput>  
loyaltyPromotionInputList)
```

## Parameters

*loyaltyPromotionInputList*

Type: List<LoyaltyManagement.CdpBasedLoyaltyPromotionInput>

Set of inputs to retrieve promotions for a member based on the member's market segment.

## Return Value

Type: List<LoyaltyManagement.CdpBasedLoyaltyPromotionOutput>

**getLoyaltyPromotions (loyaltyPromotionInputList)**

Use this method to retrieve active promotions that are eligible for a transaction journal. The action can process accrual or redemption transactions.

**Signature**

```
public static List<LoyaltyManagement.LoyaltyPromotionOutput>  
getLoyaltyPromotions (List<LoyaltyManagement.LoyaltyPromotionInput>  
loyaltyPromotionInputList)
```

**Parameters**

*loyaltyPromotionInputList*

Type: List<LoyaltyManagement.LoyaltyPromotionInput>

Set of inputs to retrieve active promotions for a transaction journal.

**Return Value**

Type: List<LoyaltyManagement.LoyaltyPromotionOutput>

**getPointsBalance (memberPointBalanceList)**

Use this method to retrieve the loyalty points balance for a loyalty program member.

**Signature**

```
public static List<LoyaltyManagement.MemberPointBalanceOutput>  
getPointsBalance (List<LoyaltyManagement.MemberPointBalanceInput> memberPointBalanceList)
```

**Parameters**

*memberPointBalanceList*

Type: List<LoyaltyManagement.MemberPointBalanceInput>

Set of inputs to get the loyalty points balance for a loyalty program member's balance.

**Return Value**

Type: List<LoyaltyManagement.MemberPointBalanceOutput>

**getTier (memberTierList)**

Use this method to retrieve the current tier for a loyalty program member.

**Signature**

```
public static List<LoyaltyManagement.MemberTierOutput>  
getTier (List<LoyaltyManagement.MemberTierInput> memberTierList)
```

## Parameters

*memberTierList*

Type: List<loyalty\_management.MemberTierInput>

Set of inputs to retrieve the current tier for a given loyalty program member.

## Return Value

Type: List<loyalty\_management.MemberTierOutput>

## **issueVoucher (issueVoucherList)**

Use this method to issue a voucher for a loyalty program member.

## Signature

```
public static List<LoyaltyManagement.IssueVoucherOutput>  
issueVoucher (List<LoyaltyManagement.IssueVoucherInput> issueVoucherList)
```

## Parameters

*issueVoucherList*

Type: List<loyalty\_management.IssueVoucherInput>

Set of inputs to issue a voucher.

## Return Value

Type: List<loyalty\_management.IssueVoucherOutput>

## **transferMemberPointsToGroups (transferMemberPointsToGroupsList)**

Use this method to transfer points from an individual member or a corporate member to the member's associated group.

## Signature

```
public static List<LoyaltyManagement.LoyaltyActionResult>  
transferMemberPointsToGroups (List<LoyaltyManagement.TransferMemberPointsToGroupsInput>  
transferMemberPointsToGroupsList)
```

## Parameters

*transferMemberPointsToGroupsList*

Type: List<loyalty\_management.TransferMemberPointsToGroupsInput>

Set of inputs to transfer points from an individual member or a corporate member.

## Return Value

Type: List<loyalty\_management.LoyaltyActionResult>

**updateProgressForCumulativePromotionUsage (updateCumulativeUsageCompletedInputList)**

Use this method to update the progress made by a loyalty program member towards attaining a Cumulative type promotion.

**Signature**

```
public static List<LoyaltyManagement.UpdateCumulativeUsageCompletedOutput>
updateProgressForCumulativePromotionUsage (List<LoyaltyManagement.UpdateCumulativeUsageCompletedInput>
updateCumulativeUsageCompletedInputList)
```

**Parameters**

*updateCumulativeUsageCompletedInputList*  
Type: List<LoyaltyManagement.UpdateCumulativeUsageCompletedInput>  
Set of inputs to update the progress for a Cumulative type promotion.

**Return Value**

Type: List<LoyaltyManagement.UpdateCumulativeUsageCompletedOutput>

## MemberPointBalanceInput Class

Use this class to retrieve loyalty points balance for a given loyalty program member. This class receives the input parameter values for the Get Member Points action.

**Namespace**

[LoyaltyManagement](#)

**Example**

```
public void pointsBalanceTest() {
    //...Create a list with LoyaltyManagement.MemberPointsBalanceInput as a datatype
    List<LoyaltyManagement.MemberPointBalanceInput> memberPointsBalanceList = new
List<LoyaltyManagement.MemberPointBalanceInput>();

    //...Create an instance of the list object and set the input values
    memberPointsBalanceList.add(new LoyaltyManagement.MemberPointBalanceInputBuilder()
        .setLoyaltyProgramMemberId('01MRM000002BD12AM')
        .setProgramName('LP01')
        .setProgramCurrencyName('Miles')
        .build());

    //...Pass the list that contains the input values to the
    LoyaltyResources.getPointsBalance() function.
    //...Store the output returned by the function in the MemberPointsBalanceOutput object

    List<LoyaltyManagement.MemberPointBalanceOutput> pointBalanceActionResults =
    LoyaltyManagement.LoyaltyResources.getPointsBalance(memberPointsBalanceList);
    System.debug('Get Member Point Balance Successful Operation: '+
```

```
pointBalanceActionResults.get(0);
    System.debug('Points balance : ' + pointBalanceActionResults.get(0).getPoints());
}
```

## MemberPointBalanceInputBuilder Class

Contains methods to build an instance of the `LoyaltyManagement.MemberPointsBalanceInput` class.

### Namespace

[LoyaltyManagement](#)

[MemberPointBalanceInputBuilder Methods](#)

### MemberPointBalanceInputBuilder Methods

The following are methods for `MemberPointBalanceInputBuilder`.

#### [build\(\)](#)

Returns an instance of the `LoyaltyManagement.MemberPointsBalanceInput` object.

#### [setLoyaltyProgramMemberId\(loyaltyProgramMemberId\)](#)

Represents the ID of the loyalty program member whose points balance is to be fetched. This is a required field.

#### [setProgramCurrencyName\(programCurrencyName\)](#)

Represents the name of the program currency. This is a required field.

#### [setProgramName\(programName\)](#)

Represents the name of the loyalty program. This is a required field.

#### **build()**

Returns an instance of the `LoyaltyManagement.MemberPointsBalanceInput` object.

#### Signature

```
public LoyaltyManagement.MemberPointBalanceInput build()
```

#### Return Value

Type: [LoyaltyManagement.MemberPointBalanceInput](#)

#### **setLoyaltyProgramMemberId(loyaltyProgramMemberId)**

Represents the ID of the loyalty program member whose points balance is to be fetched. This is a required field.

#### Signature

```
public LoyaltyManagement.MemberPointBalanceInputBuilder setLoyaltyProgramMemberId(String
loyaltyProgramMemberId)
```



### Parameters

*loyaltyProgramMemberId*  
Type: String

### Return Value

Type: [LoyaltyManagement.MemberPointBalanceInputBuilder](#)

#### **setProgramCurrencyName (programCurrencyName)**

Represents the name of the program currency. This is a required field.

### Signature

```
public LoyaltyManagement.MemberPointBalanceInputBuilder setProgramCurrencyName (String  
programCurrencyName)
```

### Parameters

*programCurrencyName*  
Type: String

### Return Value

Type: [LoyaltyManagement.MemberPointBalanceInputBuilder](#)

#### **setProgramName (programName)**

Represents the name of the loyalty program. This is a required field.

### Signature

```
public LoyaltyManagement.MemberPointBalanceInputBuilder setProgramName (String  
programName)
```

### Parameters

*programName*  
Type: String

### Return Value

Type: [LoyaltyManagement.MemberPointBalanceInputBuilder](#)

## MemberPointBalanceOutput Class

Represents the loyalty points balance for a specified loyalty program member.

## Namespace

[LoyaltyManagement](#)

[MemberPointBalanceOutput Methods](#)

## MemberPointBalanceOutput Methods

The following are methods for `MemberPointBalanceOutput`.

[getCurrencyName\(\)](#)

Returns a String that contains the name of the program currency.

[getErrors\(\)](#)

If an error occurred, returns a list providing the error code and description.

[getEscrowPointsBalance\(\)](#)

Returns a Double that contains the escrow points balance for a loyalty program member. These points are yet to be credited from Escrow to the loyalty member's points balance.

[getLoyaltyProgramMemberId\(\)](#)

Returns a String that contains the ID of the loyalty program member.

[getPoints\(\)](#)

Returns a Double that contains the points available for the loyalty program member.

[getProgramName\(\)](#)

Returns a String that contains the loyalty program name.

[isSuccess\(\)](#)

Returns a Boolean that is set to true if the member points operation was successful for this object, false otherwise.

### **getCurrencyName ()**

Returns a String that contains the name of the program currency.

### Signature

```
public String getCurrencyName ()
```

### Return Value

Type: String

### **getErrors ()**

If an error occurred, returns a list providing the error code and description.

### Signature

```
public List<LoyaltyManagement.LoyaltyActionError> getErrors ()
```

## Return Value

Type: List<[LoyaltyManagement.LoyaltyActionError](#)>

### **getEscrowPointsBalance ()**

Returns a Double that contains the escrow points balance for a loyalty program member. These points are yet to be credited from Escrow to the loyalty member's points balance.

## Signature

```
public Double getEscrowPointsBalance ()
```

## Return Value

Type: Double

### **getLoyaltyProgramMemberId ()**

Returns a String that contains the ID of the loyalty program member.

## Signature

```
public String getLoyaltyProgramMemberId ()
```

## Return Value

Type: String

### **getPoints ()**

Returns a Double that contains the points available for the loyalty program member.

## Signature

```
public Double getPoints ()
```

## Return Value

Type: Double

### **getProgramName ()**

Returns a String that contains the loyalty program name.

## Signature

```
public String getProgramName ()
```

## Return Value

Type: String

## isSuccess ()

Returns a Boolean that is set to true if the member points operation was successful for this object, false otherwise.

## Signature

```
public Boolean isSuccess ()
```

## Return Value

Type: Boolean

# MemberTierInput Class

Use this class to retrieve the current tier for a specified loyalty program member. This class receives the input parameter values for the Get Member Tier action.

## Namespace

[LoyaltyManagement](#)

## Example

```

public void getTierTest () {
    //...Create a list with LoyaltyManagement.MemberTierInput as a datatype
    List<LoyaltyManagement.MemberTierInput> memberTiersList = new
List<LoyaltyManagement.MemberTierInput> ();

    //...Create an instance of the list object and set the input values
    memberTiersList.add(new
LoyaltyManagement.MemberTierInputBuilder().setLoyaltyProgramMemberId('01MRM000002BD12AM')

        .setProgramName('LP01')
        .setTierGroupName('TG1')
        .build());

    //...Pass the list that contains the input values to the LoyaltyResources.getTier()
function.
    //...Store the output returned by the function in the MemberTierOutput object
    List<LoyaltyManagement.MemberTierOutput> memberTierActionResults =
LoyaltyManagement.LoyaltyResources.getTier(memberTiersList);
    System.debug('Get Member Tier Successful Operation:'+ memberTierActionResults.get(0));

    System.debug('Member Tier : '+ memberTierActionResults.get(0).getMemberTier());
}

```

## MemberTierInputBuilder Class

Contains methods to build an instance of the `LoyaltyManagement.MemberTierInput` class.

### Namespace

[LoyaltyManagement](#)

[MemberTierInputBuilder Methods](#)

### MemberTierInputBuilder Methods

The following are methods for `MemberTierInputBuilder`.

[build\(\)](#)

Returns an instance of the `LoyaltyManagement.MemberTierInput` object.

[setLoyaltyProgramMemberId\(loyaltyProgramMemberId\)](#)

Represents the ID of the loyalty program member whose tier details are to be fetched. This is a required field.

[setProgramName\(programName\)](#)

Represents the loyalty program name associated with the tier.

[setTierGroupName\(tierGroupName\)](#)

Represents the name of the tier group. This is a required field.

#### **build()**

Returns an instance of the `LoyaltyManagement.MemberTierInput` object.

#### Signature

```
public LoyaltyManagement.MemberTierInput build()
```

#### Return Value

Type: [LoyaltyManagement.MemberTierInput](#)

#### **setLoyaltyProgramMemberId(loyaltyProgramMemberId)**

Represents the ID of the loyalty program member whose tier details are to be fetched. This is a required field.

#### Signature

```
public LoyaltyManagement.MemberTierInputBuilder setLoyaltyProgramMemberId(String  
loyaltyProgramMemberId)
```

### Parameters

*loyaltyProgramMemberId*  
Type: String

### Return Value

Type: [LoyaltyManagement.MemberTierInputBuilder](#)

#### **setProgramName (programName)**

Represents the loyalty program name associated with the tier.

### Signature

```
public LoyaltyManagement.MemberTierInputBuilder setProgramName(String programName)
```

### Parameters

*programName*  
Type: String

### Return Value

Type: [LoyaltyManagement.MemberTierInputBuilder](#)

#### **setTierGroupName (tierGroupName)**

Represents the name of the tier group. This is a required field.

### Signature

```
public LoyaltyManagement.MemberTierInputBuilder setTierGroupName(String tierGroupName)
```

### Parameters

*tierGroupName*  
Type: String

### Return Value

Type: [LoyaltyManagement.MemberTierInputBuilder](#)

## MemberTierOutput Class

Represents the tier details for a specified loyalty program member.

### Namespace

[LoyaltyManagement](#)

[MemberTierOutput Methods](#)

## MemberTierOutput Methods

The following are methods for `MemberTierOutput`.

[getErrors\(\)](#)

If an error occurred, returns a list providing the error code and description.

[getLoyaltyProgramMemberId\(\)](#)

Returns a String that contains the loyalty program member ID.

[getMemberTier\(\)](#)

Returns a String that contains the name of the tier associated with the program member.

[getProgramName\(\)](#)

Returns a String that contains the loyalty program name associated with the tier.

[getSequenceNumber\(\)](#)

Returns an Integer that contains the ranking of the tier.

[getTierGroupName\(\)](#)

Returns a String that contains the name of the tier group.

[isSuccess\(\)](#)

Returns a Boolean that is set to true if the member tier operation was successful for this object, false otherwise.

### **getErrors ()**

If an error occurred, returns a list providing the error code and description.

### Signature

```
public List<LoyaltyManagement.LoyaltyActionError> getErrors ()
```

### Return Value

Type: List<[LoyaltyManagement.LoyaltyActionError](#)>

### **getLoyaltyProgramMemberId ()**

Returns a String that contains the loyalty program member ID.

### Signature

```
public String getLoyaltyProgramMemberId ()
```

### Return Value

Type: String

**getMemberTier()**

Returns a String that contains the name of the tier associated with the program member.

**Signature**

```
public String getMemberTier()
```

**Return Value**

Type: String

**getProgramName()**

Returns a String that contains the loyalty program name associated with the tier.

**Signature**

```
public String getProgramName()
```

**Return Value**

Type: String

**getSequenceNumber()**

Returns an Integer that contains the ranking of the tier.

**Signature**

```
public Integer getSequenceNumber()
```

**Return Value**

Type: Integer

**getTierGroupName()**

Returns a String that contains the name of the tier group.

**Signature**

```
public String getTierGroupName()
```

**Return Value**

Type: String

**isSuccess()**

Returns a Boolean that is set to true if the member tier operation was successful for this object, false otherwise.



## Signature

```
public Boolean isSuccess()
```

## Return Value

Type: Boolean

## PointsInput Class

Use this class to credit loyalty points to a loyalty program member's balance or debit loyalty points from a loyalty program member's balance. This class receives the input parameter values for the Credit and Debit point actions.

## Namespace

[LoyaltyManagement](#)

## Example

This example is to credit loyalty points to a member's balance:

```
public void creditPointsTest() {
    //...Create a list with LoyaltyManagement.PointsInput as a datatype
    List<LoyaltyManagement.PointsInput> creditPointsList = new
    List<LoyaltyManagement.PointsInput>();

    //...Create an instance of the list object and set the input values
    creditPointsList.add(new LoyaltyManagement.PointsInputBuilder()
        .setLoyaltyProgramMemberId('01MRM0000002BD12AM')
        .setProgramName('LP01')
        .setProgramCurrencyName('Miles')
        .setPoints(100)
        .setJournalId('01VRM0000004DH22AM')
        .build());

    //...Pass the list that contains the input values to the LoyaltyResources.creditPoints()
    function.
    //...Store the output returned by the function in the CreditPointsOutput object
    List<LoyaltyManagement.CreditPointsOutput> creditPointActionResults =
    LoyaltyManagement.LoyaltyResources.creditPoints(creditPointsList);
    System.debug('Credit Successful Operation:'+ creditPointActionResults.get(0));
}
```

This example is to debit loyalty points from a member's balance:

```
public void debitPointsTest() {
    //...Create a list with LoyaltyManagement.PointsInput as a datatype
    List<LoyaltyManagement.PointsInput> debitPointsList = new
    List<LoyaltyManagement.PointsInput>();

    //...Create an instance of the list object and set the input values
```

```
debitPointsList.add(new LoyaltyManagement.PointsInputBuilder()
    .setLoyaltyProgramMemberId('01MRM000002BD12AM')
    .setProgramName('LP01')
    .setProgramCurrencyName('Miles')
    .setPoints(1000)
    .setJournalId('01VRM000004COB2A2')
    .build());

//...Pass the list that contains the input values to the LoyaltyResources.debitPoints()
function.
//...Store the output returned by the function in the DebitPointsOutput object
List<LoyaltyManagement.DebitPointsOutput> debitPointActionResults =
LoyaltyManagement.LoyaltyResources.debitPoints(debitPointsList);
System.debug('Debit Successful Operation:'+ debitPointActionResults.get(0));
}
```

## PointsInputBuilder Class

Contains methods to build an instance of the `LoyaltyManagement.PointsInput` class.

### Namespace

[LoyaltyManagement](#)

[PointsInputBuilder Methods](#)

### PointsInputBuilder Methods

The following are methods for `PointsInputBuilder`.

#### [build\(\)](#)

Returns an instance of the `LoyaltyManagement.PointsInput` object.

#### [setAdditionalNotes\(additionalNotes\)](#)

Represents extra information about the transactions performed by the loyalty program member.

#### [setCorrelationId\(correlationId\)](#)

Represents the ID of any Salesforce entity.

#### [setFlowId\(flowId\)](#)

Represents the ID of the flow associated with the credit or debit points action.

#### [setJournalId\(journalId\)](#)

Represents the ID of the transaction journal. This is a required field.

#### [setLoyaltyProgramMemberId\(loyaltyProgramMemberId\)](#)

Represents the ID of the loyalty program member to whom the points must be credited or debited. This is a required field.

#### [setPoints\(points\)](#)

Represents the points available with the loyalty program member. This is a required field.

[setPointsExpirationDate\(pointsExpirationDate\)](#)

Represents the date on which fixed-model non-qualifying points credited by the action expire.

[setProgramCurrencyName\(programCurrencyName\)](#)

Represents the name of the loyalty program currency.

[setProgramName\(programName\)](#)

Represents the name of the loyalty program.

### **build()**

Returns an instance of the `LoyaltyManagement.PointsInput` object.

### Signature

```
public LoyaltyManagement.PointsInput build()
```

### Return Value

Type: [LoyaltyManagement.PointsInput](#)

### **setAdditionalNotes(additionalNotes)**

Represents extra information about the transactions performed by the loyalty program member.

### Signature

```
public LoyaltyManagement.PointsInputBuilder setAdditionalNotes(String additionalNotes)
```

### Parameters

*additionalNotes*

Type: String

### Return Value

Type: [LoyaltyManagement.PointsInputBuilder](#)

### **setCorrelationId(correlationId)**

Represents the ID of any Salesforce entity.

### Signature

```
public LoyaltyManagement.PointsInputBuilder setCorrelationId(String correlationId)
```

### Parameters

*correlationId*

Type: String

## Return Value

Type: [LoyaltyManagement.PointsInputBuilder](#)

### **setFlowId(flowId)**

Represents the ID of the flow associated with the credit or debit points action.

## Signature

```
public LoyaltyManagement.PointsInputBuilder setFlowId(String flowId)
```

## Parameters

*flowId*

Type: String

## Return Value

Type: [LoyaltyManagement.PointsInputBuilder](#)

### **setJournalId(journalId)**

Represents the ID of the transaction journal. This is a required field.

## Signature

```
public LoyaltyManagement.PointsInputBuilder setJournalId(String journalId)
```

## Parameters

*journalId*

Type: String

## Return Value

Type: [LoyaltyManagement.PointsInputBuilder](#)

### **setLoyaltyProgramMemberId(loyaltyProgramMemberId)**

Represents the ID of the loyalty program member to whom the points must be credited or debited. This is a required field.

## Signature

```
public LoyaltyManagement.PointsInputBuilder setLoyaltyProgramMemberId(String  
loyaltyProgramMemberId)
```

## Parameters

*loyaltyProgramMemberId*

Type: String

## Return Value

Type: [LoyaltyManagement.PointsInputBuilder](#)

### **setPoints (points)**

Represents the points available with the loyalty program member. This is a required field.

## Signature

```
public LoyaltyManagement.PointsInputBuilder setPoints(Double points)
```

## Parameters

*points*

Type: Double

## Return Value

Type: [LoyaltyManagement.PointsInputBuilder](#)

### **setPointsExpirationDate (pointsExpirationDate)**

Represents the date on which fixed-model non-qualifying points credited by the action expire.

## Signature

```
public LoyaltyManagement.PointsInputBuilder setPointsExpirationDate(String pointsExpirationDate)
```

## Parameters

*pointsExpirationDate*

Type: String

The supported date format is YYYY-MM-DD. This input is available in API version 55.0 and later.

## Return Value

Type: [LoyaltyManagement.PointsInputBuilder](#)

### **setProgramCurrencyName (programCurrencyName)**

Represents the name of the loyalty program currency.

## Signature

```
public LoyaltyManagement.PointsInputBuilder setProgramCurrencyName(String programCurrencyName)
```

## Parameters

*programCurrencyName*  
Type: String

## Return Value

Type: [LoyaltyManagement.PointsInputBuilder](#)

### **setProgramName (programName)**

Represents the name of the loyalty program.

## Signature

```
public LoyaltyManagement.PointsInputBuilder setProgramName(String programName)
```

## Parameters

*programName*  
Type: String

## Return Value

Type: [LoyaltyManagement.PointsInputBuilder](#)

# TransferMemberPointsToGroupsInput Class

Use this class to transfer points from an individual member or a corporate member to the member's associated group. This class receives the input parameter values for the Transfer Member Points To Groups action.

## Namespace

[LoyaltyManagement](#)

## Example

```
public static void transferPointsToGroupsTest () {  
  
    //Create a list with LoyaltyManagement.TransferMemberPointsToGroupsInput as a datatype  
  
    List<LoyaltyManagement.TransferMemberPointsToGroupsInput> transferPointsList =  
        new List<LoyaltyManagement.TransferMemberPointsToGroupsInput> ();  
  
    //Create an instance of transferPointsList to set the transaction journal ID as input  
    transferPointsList.add(new LoyaltyManagement.TransferMemberPointsToGroupsInputBuilder ()  
  
        .setJournalId('01VT10000008Cn1MAE')  
        .build());  
}
```

```
//Pass the list to the LoyaltyResources.transferMemberPointsToGroups() interface.
//Store the output returned by the interface in an instance of the 'LoyaltyActionResult'
object
List<LoyaltyManagement.LoyaltyActionResult> transferResults =
    LoyaltyManagement.LoyaltyResources.transferMemberPointsToGroups (transferPointsList);

//Return the status of the Apex call
System.debug ('Transfer points to groups is successful:'+
transferResults.get (0) .isSuccess ());
}
```

## TransferMemberPointsToGroupsInputBuilder Class

Contains methods to build an instance of the `LoyaltyManagement.TransferMemberPointsToGroupsInput` class.

### Namespace

[LoyaltyManagement](#)

[TransferMemberPointsToGroupsInputBuilder Methods](#)

### TransferMemberPointsToGroupsInputBuilder Methods

The following are methods for `TransferMemberPointsToGroupsInputBuilder`.

[build\(\)](#)

Returns an instance of the `LoyaltyManagement.TransferMemberPointsToGroupsInput` object.

[setJournalId\(journalId\)](#)

Represents the transaction journal that is used to transfer points to groups. This is a required field.

#### **build()**

Returns an instance of the `LoyaltyManagement.TransferMemberPointsToGroupsInput` object.

#### Signature

```
public LoyaltyManagement.TransferMemberPointsToGroupsInput build()
```

#### Return Value

Type: [LoyaltyManagement.TransferMemberPointsToGroupsInput](#)

#### **setJournalId(journalId)**

Represents the transaction journal that is used to transfer points to groups. This is a required field.

## Signature

```
public LoyaltyManagement.TransferMemberPointsToGroupsInputBuilder setJournalId(String
journalId)
```

## Parameters

*journalId*

Type: String

The transaction journal must be associated with either an Individual or a Corporate type member. The points are transferred to the member's associated groups.

## Return Value

Type: [LoyaltyManagement.TransferMemberPointsToGroupsInputBuilder](#)

# UpdateCumulativeUsageCompletedInput Class

Use this class to update the progress made by a loyalty program member towards attaining a Cumulative type promotion. The action creates a Loyalty Program Member Promotion record if the promotion isn't associated with the member. This class receives the input parameter values for the Update Cumulative Usage Completed action.

## Namespace

[LoyaltyManagement](#)

## Example

```
public static void updateProgressForCumulativePromotionUsageTest() {

    //Create a list with LoyaltyManagement.UpdateCumulativeUsageCompletedInput as a datatype

    List<LoyaltyManagement.UpdateCumulativeUsageCompletedInput>
updateCumulativeUsageCompletedInputList =
    new List<LoyaltyManagement.UpdateCumulativeUsageCompletedInput>();

    //Create an instance of LoyaltyManagement.UpdateCumulativeUsageCompletedInput class and
set the inputs
    LoyaltyManagement.UpdateCumulativeUsageCompletedInput input = new
LoyaltyManagement.UpdateCumulativeUsageCompletedInput('01Mxx00000000EXEAY',
'0c8xx00000000BJAAY', 200);
    LoyaltyManagement.UpdateCumulativeUsageCompletedInput input1 = new
LoyaltyManagement.UpdateCumulativeUsageCompletedInput('01Mxx00000000G9EAI',
'0c8xx00000000BJAAY', 200);
    updateCumulativeUsageCompletedInputList.add(input);
    updateCumulativeUsageCompletedInputList.add(input1);

    //Store the output returned by the interface in an instance of the
UpdateCumulativeUsageCompletedOutput object
    List<LoyaltyManagement.UpdateCumulativeUsageCompletedOutput> results =
```



```
LoyaltyManagement.LoyaltyResources.updateProgressForCumulativePromotionUsage(updateCumulativeUsageCompletedInputList);

//Return the status of the Apex call
System.debug('updateProgressForCumulativePromotionUsage Successful Operation:'+
results.get(0).isSuccess());
}
```

[UpdateCumulativeUsageCompletedInput Constructors](#)

## UpdateCumulativeUsageCompletedInput Constructors

The following are constructors for `UpdateCumulativeUsageCompletedInput`.

[UpdateCumulativeUsageCompletedInput\(loyaltyProgramMemberId, promotionId, usageIncrementForCumulativePromotion\)](#)

Creates an instance of the `LoyaltyManagement.UpdateCumulativeUsageCompletedInput` class.

**UpdateCumulativeUsageCompletedInput(loyaltyProgramMemberId, promotionId, usageIncrementForCumulativePromotion)**

Creates an instance of the `LoyaltyManagement.UpdateCumulativeUsageCompletedInput` class.

### Signature

```
public UpdateCumulativeUsageCompletedInput(String loyaltyProgramMemberId, String
promotionId, Integer usageIncrementForCumulativePromotion)
```

### Parameters

*loyaltyProgramMemberId*

Type: String

The ID of the loyalty program member who progress towards attaining a cumulative promotion must be updated. This is a required field.

*promotionId*

Type: String

The ID of the Cumulative type promotion for which member's progress must be updated. This is a required field.

*usageIncrementForCumulativePromotion*

Type: Integer

The value or quantity that must be added to member's usage towards attaining the promotion. This is a required field.

## UpdateCumulativeUsageCompletedOutput Class

Represents the details of the progress made by a loyalty program member towards attaining a Cumulative type promotion.

## Namespace

[LoyaltyManagement](#)

[UpdateCumulativeUsageCompletedOutput Methods](#)

## UpdateCumulativeUsageCompletedOutput Methods

The following are methods for `UpdateCumulativeUsageCompletedOutput`.

[getCumulativeUsageTarget\(\)](#)

Returns the target value or quantity which members must achieve to attain the promotion.

[getErrors\(\)](#)

If an error occurred, returns a list providing the error code and description.

[getUpdatedCumulativeUsageCompleted\(\)](#)

Returns the updated value or quantity by which the member has progressed towards attaining the promotion.

[isSuccess\(\)](#)

Returns a Boolean that is set to true if the update cumulative usage completed operation was successful for this object, false otherwise.

### **getCumulativeUsageTarget ()**

Returns the target value or quantity which members must achieve to attain the promotion.

### Signature

```
public Integer getCumulativeUsageTarget ()
```

### Return Value

Type: Integer

### **getErrors ()**

If an error occurred, returns a list providing the error code and description.

### Signature

```
public List<LoyaltyManagement.LoyaltyActionError> getErrors ()
```

### Return Value

Type: List<[LoyaltyManagement.LoyaltyActionError](#)>

### **getUpdatedCumulativeUsageCompleted ()**

Returns the updated value or quantity by which the member has progressed towards attaining the promotion.

### Signature

```
public Integer getUpdatedCumulativeUsageCompleted()
```

### Return Value

Type: Integer

### **isSuccess ()**

Returns a Boolean that is set to true if the update cumulative usage completed operation was successful for this object, false otherwise.

### Signature

```
public Boolean isSuccess()
```

### Return Value

Type: Boolean

## CHAPTER 6 Loyalty Management Business APIs

### In this chapter ...

- REST Reference
- Loyalty Management Connect API Interchangeable Request Fields

Loyalty Management Business APIs are RESTful APIs that are sometimes available as Apex classes and methods.

## REST Reference

---

You can access Loyalty Management APIs using REST endpoints. These REST APIs follow similar conventions as Connect REST APIs.

To understand the architecture, authentication, rate limits, and how the requests and responses work, see [Connect REST API Developer Guide](#).

### [Resources](#)

Here's a list of Loyalty Management API resources.

### [Request Bodies](#)

Here's a list of Loyalty Management API request bodies.

### [Response Bodies](#)

Here's a list of Loyalty Management API response bodies.

## Resources

Here's a list of Loyalty Management API resources.

### [Corporate Member Enrollments](#)

Enroll corporate members in a loyalty program. Create a business account and a Corporate type loyalty program member record for the new member.

### [Individual Member Enrollments](#)

Enroll individual members in a loyalty program. Create a person account and an Individual type loyalty program member record for the new member.

### [Member Benefits](#)

Get benefits for a loyalty program member.

### [Member Profile](#)

Get the details of a loyalty program member.

### [Member Engagement Trail](#)

Fetches current and target values of the milestones defined in an engagement trail and the rewards received after achieving milestones.

### [Member Vouchers](#)

Get vouchers that have been issued to a loyalty program member.

### [Redeem Voucher](#)

Redeem a voucher for a loyalty program member. The status of the voucher gets updated to redeemed and the use date gets updated to the current date.

### [Transaction History](#)

Get transaction journals of a loyalty member belonging to a loyalty program. Obtain either accrual or redemption transaction journal records.

### [Transaction Ledger Summary](#)

Retrieve an aggregate view on the transaction journal and the loyalty ledger records.

### [Transaction Journals Execution](#)

Use this resource to create and process a list of transaction journals against a corresponding loyalty program process.

[Transaction Journals Simulation](#)

Runs a simulation to determine how transaction journals are processed against their corresponding loyalty program processes. When transaction journals meet the criteria and conditions for a program process, actions that are set up in the process are triggered for the transaction journals.

[Loyalty Program Process](#)

Run a loyalty process created from a loyalty program process template.

[Gamification](#)

Who doesn't like to play a game? Engage your customers more by gamifying customer activities, such as placing an order, or events, such as birthdays and anniversaries.

[Promotion](#)

The promotion APIs allow you to set eligibility rules and limits for a promotion, choose a promotion template, and create a promotion. You can also get the details of a specific promotion, and view a list of eligible promotions for a customer's cart.

## Corporate Member Enrollments

Enroll corporate members in a loyalty program. Create a business account and a Corporate type loyalty program member record for the new member.

To use this resource, Person Accounts must be enabled in your org. The profile of the user who executes the resource must have Business Account selected as the default record type for business accounts. If the profile of the user has a different record type selected as the default record type for business accounts, an Account record of the selected record type is created for the member.

### Resource

```
/loyalty-programs/${loyaltyProgramName}/corporate-member-enrollments
```

### Example

```
https://yourInstance.salesforce.com/services/data/vXX.X  
/loyalty-programs/program1/corporate-member-enrollments
```

### Available version

55.0

### Requires Chatter

No

### HTTP methods

POST

### Request body for POST

#### Root XML tag

```
<corporateMemberEnrollment>
```

#### JSON example

```
{
  "enrollmentDate" : "2021-01-01T05:05:05",
  "membershipNumber" : "Member123",
  "associatedAccountDetails" : {
    "name" : "Name1",
    "phone" : "908989898",
    "website" : "www.ab.com",
    "allowDuplicateRecords" : "false",
```

```

    "additionalAccountFieldValues" : {
      "attributes" : {
        "CustomNumber__c" : "20.0"
      }
    },
    "memberStatus" : "Active",
    "referredBy" : "Member12",
    "referredByMemberReferralCode" : "MPX12adf",
    "createTransactionJournals" : "true",
    "transactionJournalStatementFrequency" : "Monthly",
    "additionalMemberFieldValues" : {
      "attributes" : {
        "CustomBoolean__c" : "true"
      }
    }
  }
}

```

### Properties

Name	Type	Description	Required or Optional	Available Version
additionalMemberFieldValues	Map<String, String>	The custom attributes of the member in the loyalty program member record. <ul style="list-style-type: none"> <li>• Keys</li> <li>• Values</li> </ul>	Optional	55.0
associatedAccountDetails	<a href="#">Loyalty Account Input</a>	The business account details of the member.	Required	55.0
canReceivePartnerPromotions	Boolean	Indicates whether the member can receive partner promotions for the program ( <code>true</code> ) or not ( <code>false</code> ).	Optional	55.0
canReceivePromotions	Boolean	Indicates whether the member can receive promotions for the program.	Optional	55.0
createTransactionJournals	Boolean	Indicates whether to create the transaction journal records ( <code>true</code> ) or not ( <code>false</code> ): The transaction journal records are created to: <ul style="list-style-type: none"> <li>• Credit points to the member for enrolling.</li> <li>• Credit points to the member who has referred the new member.</li> </ul>	Optional	55.0

Name	Type	Description	Required or Optional	Available Version
enrollmentChannel	String	The channel used by members to enroll into a loyalty program.  Possible values are: <ul style="list-style-type: none"> <li>• CallCenter</li> <li>• Email</li> <li>• Franchise</li> <li>• Mobile</li> <li>• Partner</li> <li>• Pos</li> <li>• Print</li> <li>• Social</li> <li>• Store</li> <li>• Web</li> </ul>	Optional	55.0
enrollmentDate	String	The date on which a member is enrolled.	Required	55.0
memberStatus	String	Status of the member. Possible values are: <ul style="list-style-type: none"> <li>• Active</li> <li>• Inactive</li> <li>• Custom value</li> </ul>	Required	55.0
membershipEndDate	String	The date on which the loyalty program member's membership ends.	Optional	55.0
membershipNumber	String	The unique membership number of the loyalty program member.	Required	55.0
referredBy	String	The ID of the loyalty program member who has referred the member.	Optional	55.0
referrerReferralCode	String	The referral code of the loyalty program member who has referred the member.	Optional	57.0
transactionJournalStatementFrequency	String	The frequency at which transaction journal statements must be delivered to the member. Possible values are: <ul style="list-style-type: none"> <li>• Daily</li> <li>• Monthly</li> <li>• Yearly</li> </ul>	Optional	55.0



Name	Type	Description	Required or Optional	Available Version
transactionJournalStatementMethod	String	The method used to deliver transaction journal statements to the member.  Possible values are: <ul style="list-style-type: none"> <li>Email</li> <li>Mail</li> </ul>	Optional	55.0

**Response body for POST**

[Corporate Member Enrollment Output](#)

## Individual Member Enrollments

Enroll individual members in a loyalty program. Create a person account and an Individual type loyalty program member record for the new member.

To use this resource, Person Accounts must be enabled in your org. The profile of the user who executes the resource must have Person Account selected as the default record type for person accounts. If the profile of the user has a different record type selected as the default record type for person accounts, an Account record of the selected record type is created for the member.

**Resource**

```
/loyalty-programs/${loyaltyProgramName}/individual-member-enrollments
```

**Example**

```
https://yourInstance.salesforce.com/services/data/vXX.X  
/loyalty-programs/program1/individual-member-enrollments
```

**Available version**

55.0

**Requires Chatter**

No

**HTTP methods**

POST

**Request body for POST****Root XML tag**

```
<individualMemberEnrollment>
```

**JSON example**

```
{
  "enrollmentDate" : "2021-01-01T00:00:00",
  "membershipNumber" : "Member1",
  "associatedContactDetails" : {
    "firstName" : "Name1",
    "lastName" : "Name2",
    "email" : "ab@email.com",
    "allowDuplicateRecords" : "false",
```

```

    "additionalContactFieldValues" : {
      "attributes" : {
        "PersonMailingCity" : "Delhi",
        "CustomNumber__c" : "20.0"      }
    }
  },

  "memberStatus" : "Active",
  "referredBy" : "Member2",
  "referredByMemberReferralCode" : "MPX12adf",
  "createTransactionJournals" : "true",
  "transactionJournalStatementFrequency" : "Monthly",
  "transactionJournalStatementMethod" : "Mail",
  "enrollmentChannel" : "Email",
  "canReceivePromotions": "true",
  "canReceivePartnerPromotions" : "true",
  "membershipEndDate" : "2022-01-01T00:00:00",
  "relatedCorporateMembershipNumber" : "Member3",
  "additionalMemberFieldValues" : {
    "attributes" : {
      "CustomBoolean__c" : "true"      }
  }
}

```

### Properties

Name	Type	Description	Required or Optional	Available Version
additionalMemberFieldValues	Map<String, String>	The custom attributes of the member in the loyalty program member record. <ul style="list-style-type: none"> <li>• Keys</li> <li>• Values</li> </ul>	Required	55.0
associatedContact on page 367 Details	<a href="#">Loyalty Contact Input</a> on page 367	The person account record details of the individual type member.	Required	55.0
canReceivePartnerPromotions	Boolean	Indicates whether the member can receive partner promotions for the program ( <code>true</code> ) or not ( <code>false</code> ).	Optional	55.0
canReceivePromotions	Boolean	Indicates whether the loyalty program member can receive promotions.	Optional	55.0
createTransactionJournals	Boolean	Indicates whether to create the transaction journal records ( <code>true</code> ) or not ( <code>false</code> ):  The transaction journal records are created to: <ul style="list-style-type: none"> <li>• Credit points to the member for enrolling.</li> </ul>	Optional	55.0

Name	Type	Description	Required or Optional	Available Version
		<ul style="list-style-type: none"> <li>Credit points to the member who has referred the new member.</li> </ul>		
enrollmentChannel	String	<p>The channel used to by loyalty program member to enroll in to the loyalty program. Possible values are:</p> <ul style="list-style-type: none"> <li>Pos</li> <li>Web</li> <li>Email</li> <li>CallCenter</li> <li>Social</li> <li>Mobile</li> <li>Store</li> <li>Franchise</li> <li>Partner</li> <li>Print</li> </ul>	Optional	55.0
enrollmentDate	Date	<p>The date on which the loyalty program members are enrolled in to the loyalty program.</p> <p>The format is: 2020-09-08T00:00:00</p>	Required	55.0
memberStatus	String	<p>The status of the loyalty program member. Possible values are:</p> <ul style="list-style-type: none"> <li>Active</li> <li>Inactive</li> <li>Custom</li> </ul>	Required	55.0
membershipEndDate	String	The date on which the loyalty program member's membership ends.	Optional	55.0
membershipNumber	String	The unique membership number of the loyalty program member.	Required	55.0
referredBy	String	The membership number of the loyalty program member who has referred the member.	Optional	55.0
referralCode	String	The referral code of the loyalty program member who has referred the member.	Optional	57.0
relatedCorporateMembershipNumber	String	External corporate membership number of the member.	Optional	55.0

Name	Type	Description	Required or Optional	Available Version
transactionJournalStatementFrequency	String	The frequency at which transaction journal statements must be delivered to the member. Possible values are: <ul style="list-style-type: none"> <li>Monthly</li> <li>Quarterly</li> </ul>	Optional	55.0
transactionJournalStatementMethod	String	The method used to deliver transaction journal statements to the member. Possible values are: <ul style="list-style-type: none"> <li>Mail</li> <li>Email</li> </ul>	Optional	55.0

**Response body for POST**[Individual Member Enrollment Output](#)

## Member Benefits

Get benefits for a loyalty program member.

**Resource**

```
/connect/loyalty/member/{memberId}/memberbenefits
```

**Resource example**

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/loyalty/member/01M0000A0th124242af/memberbenefits
```

**Available version**

51.0

**Requires Chatter**

No

**HTTP methods**

GET

**Response body**[Member Benefits Output](#)

## Member Profile

Get the details of a loyalty program member.

**Resource**

```
/loyalty-programs/{loyaltyProgramName}/members
```

**Examples**

```
https://yourInstance.salesforce.com/services/data/v63.0/loyalty-programs/{baLoyalty}/members
```

```
https://yourInstance.salesforce.com/services/data/v63.0/loyalty-programs/{baLoyalty}/members?memberId=007
```

```
https://yourInstance.salesforce.com/services/data/v63.0/loyalty-programs/{baLoyalty}/members?membershipNumber=doc_member1
```

```
https://yourInstance.salesforce.com/services/data/v63.0/loyalty-programs/{baLoyalty}/members?programCurrencyName=Yen
```

**Available version**

55.0



**Requires Chatter**

No

**HTTP methods**

GET

**Request parameters for GET**

Parameter Name	Type	Description	Required or Optional	Available Version
memberId	String	The ID of the loyalty program member.	Required	55.0
			 <b>Note:</b> This field is optional if you provide the membership number.	
membershipNumber	String	The membership number of the loyalty program member.	Required	55.0
			 <b>Note:</b> This field is optional if you provide the member ID.	
programCurrencyName	String	The name of the loyalty program currency associated with the member.  Use this parameter to get the details of the member's points-related information for a specific currency.	Optional	55.0

**Response body for GET**[Member Profile Output](#)

## Member Engagement Trail

Fetches current and target values of the milestones defined in an engagement trail and the rewards received after achieving milestones.

### Resource

```
/services/data/vXX.X/loyalty/programs/{programName}/members/{membershipNumber}/engagement-trail?$promotionId={promotionId}
```

### Examples

```
https://yourInstance.salesforce.com/services/data/vXX.X
/loyalty/programs/{programName}/members/{membershipNumber}/engagement-trail?$promotionId={promotionId}
```

```
https://yourInstance.salesforce.com/services/data/vXX.X
/loyalty/programs/{programName}/members/{membershipNumber}/engagement-trail?$promotionId={promotionId}
```

### Available version

62.0

### Requires Chatter

No

### HTTP methods

GET

### Path Parameters

Parameter Name	Type	Description	Required or Optional	Available Version
programName	String	Name of the loyalty program.	Required	62.0
membershipNumber	String	The membership number of the loyalty program member.	Required	62.0

### Query Parameters

Parameter Name	Type	Description	Required or Optional	Available Version
promotionId	String	The ID of the promotion.	Required	62.0

### Response Body

[Member Engagement Trail Output](#) on page 442

## Member Vouchers

Get vouchers that have been issued to a loyalty program member.

### Resource

```
/loyalty/programs/{programName}/members/{membershipNumber}/vouchers
```

**Resource Examples**

```
https://yourInstance.salesforce.com/services/data/v58.0
/connect/loyalty/programs/NTO/members/M1/vouchers
```

```
https://yourInstance.salesforce.com/services/data/v58.0
/connect/loyalty/programs/NTO/members/M1/vouchers?voucherStatus=Cancelled,Expired
```

```
https://yourInstance.salesforce.com/services/data/v58.0
/connect/loyalty/programs/NTO/members/M1/vouchers?pageNumber=2&voucherStatus=Issued&productName=Linea,Creta
```

**Available version**

58.0





**Requires Chatter**


No

**HTTP methods**

GET

**Query parameters**

Parameter Name	Type	Description	Required or Optional	Available Version
pageNumber	Integer	The page number to be returned. If you don't specify a value, the first page is returned.   <b>Note:</b> Each page contains 200 vouchers, sorted by their date of creation.	Optional	58.0
productCategoryId	String	The ID of a product category associated with the member vouchers to be retrieved.   <b>Note:</b> You can specify up to 20 product category IDs.	Optional	58.0
productCategoryName	String	The product category name associated with the vouchers to be retrieved.   <b>Note:</b> You can specify up to 20 product categories.	Optional	58.0
productId	String	The product ID associated with the vouchers to be retrieved.   <b>Note:</b> You can specify up to 20 product IDs.	Optional	58.0

Parameter Name	Type	Description	Required or Optional	Available Version
productName	String	The product name associated with the vouchers to be retrieved.   <b>Note:</b> You can specify up to 20 product names.	Optional	58.0
voucherStatus	String	The status of the member vouchers to filter the API response. Possible values are: <ul style="list-style-type: none"> <li>Cancelled</li> <li>Expired</li> <li>Issued</li> <li>Redeemed</li> </ul>	Optional	58.0
sortBy	String	The voucher field for sorting the API response. The default value is <code>ExpirationDate</code> . Possible values are: <ul style="list-style-type: none"> <li>CreatedDate</li> <li>EffectiveDate</li> <li>ExpirationDate</li> </ul>	Optional	58.0
sortOrder	String	The sort order of the API response. The default value is <code>Ascending</code> . Possible values are: <ul style="list-style-type: none"> <li>Ascending</li> <li>Descending</li> </ul>	Optional	58.0

**Response body for GET**[Vouchers List](#)**Redeem Voucher**

Redeem a voucher for a loyalty program member. The status of the voucher gets updated to redeemed and the use date gets updated to the current date.

**Resource**

```
/services/data/vXX.X/loyalty/programs/{programName}/members/{membershipNumber}/vouchers/{voucherCode}/redeem
```



**Examples**

```
https://yourInstance.salesforce.com/services/data/vXX.X
/loyalty/programs/DemoProgram/members/M1234/vouchers/VC001/redeem
```

```
https://yourInstance.salesforce.com/services/data/vXX.X
/loyalty/programs/DemoProgram/members/M1234/vouchers/VC001/redeem?voucherNumber=00000006
```

**Available version**

51.0



**Requires Chatter**

No

**HTTP methods**

POST

**Parameters for POST**

Parameter Name	Type	Description	Required or Optional	Available Version
membershipNumber	String	The loyalty program membership number.	Required	51.0
programName	String	The loyalty program name.	Required	51.0
voucherCode	String	The voucher code to be redeemed.  <b>Note:</b> The voucher definition associated with the voucher code must be active to redeem the voucher.	Required	51.0
voucherNumber	String	The auto-generated voucher number.  <b>Note:</b> Use this query parameter if multiple vouchers exist with the same voucher code.	Optional	55.0

**Request body for POST**


The request body is available in version 53.0 and later.

**JSON example**


```
{
  "contactID": "003xx000004WjzAAAS",
  "action": "Reserve",
  "reservationKey": "key1",
  "reservedValue": 50.57,
  "redeemAmount": 50.57,
  "currencyIsoCode": "Yen"
}
```

JSON example to redeem vouchers using `contactId` or `accountID`.

```
{
  "redemptionAmount": "12",
  "contactId": "003xx000004WkhEAS",
  "accountId": "001xx000003H3NnAAK",
  "memberId": "01Mxx00000002HxEAI"
}
```

 **Note:** Provide `contactId` to redeem the voucher. If `accountID` is provided, then the associated `contactId` is fetched and the voucher is redeemed

### Properties

Name	Type	Description	Required or Optional	Available Version
<code>contactID</code>	String	ID of the contact associated with the account.	Optional	62.0
<code>currencyIsoCode</code>	String	Three-letter ISO 4217 currency code of the monetary currency in which the voucher is redeemed.	Optional	55.0
<code>redeemAmount</code>	Double	<p>The value of the voucher to be redeemed. Specify a value when a member hasn't redeemed the full value of the voucher. This value is only applicable for vouchers issued using Fixed Value voucher definitions.</p> <p>A value isn't required when the member has redeemed the complete value of the voucher in a single transaction.</p> <p> <b>Note:</b> If you don't specify a value to be redeemed for the fixed-value voucher, the voucher status changes to <code>Redeemed</code>, and the remaining value becomes zero.</p>	Optional	53.0
<code>action</code>	ReservationAction	Specify whether to redeem, reserve, or reinstate the voucher. The values for this property could be <code>Reserve</code> , <code>Reinstate</code> , or <code>Redeem</code> .	Required	62.0
<code>reservedValue</code>	Currency	Specify the amount reserved for partially redeemable vouchers.	Optional	62.0
<code>reservationKey</code>	String	Specify the key used to redeem or reinstate a reserved voucher.	Optional	62.0

**Response body for POST**

[Redeem Voucher Output](#) on page 414 (available in version 53.0 and later)

This table lists HTTP response code descriptions that are unique to this resource.

HTTP Response Code	Error Code	Description
400	INVALID_API_INPUT	The voucher is expired. Specify a voucher that hasn't expired.
400	INVALID_API_INPUT	The voucher is canceled. Specify a voucher that hasn't canceled.
400	INVALID_API_INPUT	Specify a valid voucher code and try again.
400	INVALID_API_INPUT	Specify a valid loyalty program name and try again.
400	INVALID_API_INPUT	Specify a valid loyalty program membership number and try again.
400	INVALID_API_INPUT	The voucher definition is inactive. Specify a voucher that's associated with an active voucher definition.
409	ALREADY_REDEEMED_VOUCHER	The voucher is redeemed. Specify a voucher that hasn't redeemed.
500	MULTIPLE_VOUCHERS	Duplicate vouchers for this voucher code. Either specify a different voucher code or ask your Salesforce admin for help.
500	UNKOWN_EXCEPTION	The voucher redemption failed. Try again or ask your Salesforce admin for help.

**SEE ALSO:**

[Connect REST API Developer Guide: Status Codes and Error Responses](#)

## Transaction History

Get transaction journals of a loyalty member belonging to a loyalty program. Obtain either accrual or redemption transaction journal records.

**Resource**

```
/connect/loyalty/programs/{programName}/transaction-history
```

**Example**

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/loyalty/programs/NTO/transaction-history?page=2
```

**Available version**

56.0

**Requires Chatter**


No

**HTTP methods**

POST

**Request parameters for POST**

Parameter Name	Type	Description	Required or Optional	Available Version
programName	String	Name of the loyalty program.	Required	56.0
page	Integer	Number of the page you want returned. If you don't specify a value, the first page is returned.	Optional	56.0

 **Note:** By default, each page contains 200 records.

**Request body for POST****JSON example**

```
{
  "membershipNumber": "doc_member1",
  "journalType": "Accrual",
  "journalSubType": "Social",
  "periodStartDate": "2022-01-27T00:00:00Z",
  "periodEndDate": "2022-11-27T00:00:00Z"
}
```

**Properties**

Name	Type	Description	Required or Optional	Available Version
journalSubType	String	The subtype of the transaction journal record.	Optional	56.0
journalType	String	The type of the transaction journal. Possible values include: <ul style="list-style-type: none"> <li>• Accrual</li> <li>• Redemption</li> </ul>	Required	56.0
membershipNumber	String	The membership number of the loyalty program member.	Required	56.0
periodEndDate	String	Retrieve transaction journals until this date.	Optional	56.0
periodStartDate	String	Retrieve transaction journals starting on this date.	Required	56.0

**Response body for POST**

[Loyalty View Transaction Journal History Output](#)

## Transaction Ledger Summary

Retrieve an aggregate view on the transaction journal and the loyalty ledger records.

### Resource

```
/loyalty/programs/{programName}/members/{membershipNumber}/transaction-ledger-summary
```

### Resource Examples

```
https://yourInstance.salesforce.com/services/data/v63.0/loyalty/programs/NTO/members/M1/transaction-ledger-summary
```

```
https://yourInstance.salesforce.com/services/data/v63.0/loyalty/programs/NTO/members/M1/transaction-ledger-summary?journalSubType={provide_value}
```

```
https://yourInstance.salesforce.com/services/data/v63.0/loyalty/programs/NTO/members/M1/transaction-ledger-summary?page={provide_value}&journalType={provide_value}&journalSubType={provide_value}
```

### Available version

58.0


### Requires Chatter

No

### HTTP methods

GET

### Query parameters for GET

Parameter Name	Type	Description	Required or Optional	Available Version
journalSubTypeName	String	Returns the transaction journals filtered by the specified journal subtype.	Optional	58.0
allTransactionFields	String	Returns the list of transaction journal fields.	Optional	58.0
journalTypeName	String	Returns the transaction journals filtered by the specified journal type.	Optional	58.0
pageNumber	Integer	Returns the transaction journals for the specified page number.   <b>Note:</b> The API returns the first page by default. Each page contains 200 transaction journals, which are sorted according to the date the transaction journal record was created.	Optional	58.0
periodEndDate	String	Returns the transaction journals filtered by the specified end date.	Optional	58.0
periodStartDate	String	Returns the transaction journals filtered by the specified start date.	Optional	58.0

### Response body for GET

[Transaction Journal Summary Output](#)

## Transaction Journals Execution

Use this resource to create and process a list of transaction journals against a corresponding loyalty program process.

The loyalty program processes determine how transaction journals are processed. When transaction journals meet the criteria and conditions for a program process, actions that are set up in the process are triggered for the transaction journals. For more information on how to set up program process, see [LoyaltyProgramSetup](#).

### Special Access Rules

To use this resource, your org must have either B2C - Loyalty, B2C - Loyalty Plus, Loyalty Management - Growth, or Loyalty Management - Advanced license enabled, and you must be assigned the Loyalty Management permission set.

### Resource

```
/connect/realtime/loyalty/programs/${programName}
```

### Resource example

```
https://yourInstance.salesforce.com/services/data/vXX.X/connect/realtime/loyalty/programs/CloudKicksInnerCircle
```

### Available version

54.0

### Requires Chatter

No

### HTTP methods

POST

### Request body for POST

#### JSON example

In the JSON, you can either provide the details of transaction journals that aren't available in the org, or a list of IDs that belong to transaction journals available in the org. If you provide the details of transaction journals that aren't available in the org, each transaction journal's applicable loyalty program process applies the eligible rules, executes the applicable actions, and then creates the transaction journal.

Using the transaction journal details:



**Note:** Some of the API request fields contain field names that are interchangeable. For information on the list of interchangeable field names, see [Loyalty Management Connect API Interchangeable Request Fields](#).

Here's a JSON example that creates a transaction journal for a purchase-related transaction made by a loyalty program member of a retail company. It has all the transaction journal-related information along with the details of the promotion applied, including the different types of rewards the transaction is eligible for. For information on the TransactionJournal object fields, see [https://developer.salesforce.com/docs/atlas.en-us.loyalty.meta/loyalty/sforce\\_api\\_objects\\_transactionjournal.htm](https://developer.salesforce.com/docs/atlas.en-us.loyalty.meta/loyalty/sforce_api_objects_transactionjournal.htm).

```
{
  "transactionJournals": [
    {
      "ActivityDate": "2023-11-27T12:45:19Z",
      "JournalTypeId": "01ERM00000001Bb2AI",
      "LoyaltyProgramId": "01pRM00000002YTYAY",
      "MemberId": "01MRM00000002Fmu2AE",
      "Status": "Pending",
      "ProductId": "01txx00000006iTsAAI",
      "ProductCategoryId": "0ZSSB00000002d8X4AQ",
      "Quantity": "4",
    }
  ]
}
```

```

"TransactionAmount": "500",
"appliedPromotions": [
  {
    "promotionId": "0c8RM0000004FiXYAU",
    "rewards": [
      {
        "rewardType": "Discount",
        "discountAmount": 15
      },
      {
        "rewardType": "Points",
        "loyaltyProgramCurrencyName": "Coins",
        "points": 100,
        "relatedInformation": "Test",
        "notes": "points reward"
      },
      {
        "rewardType": "Voucher",
        "voucherDefinitionName": "10% Discount Voucher",
        "voucherExpirationDate": "2023-12-15",
        "voucherEffectiveDate": "2023-11-29",
        "notes": "discount voucher"
      },
      {
        "rewardType": "Badge",
        "loyaltyProgramBadgeName": "Gen AI Badge",
        "memberBadgeStatus": "Active",
        "badgeValidityEndDate": "2023-12-15",
        "reason": "purchase"
      },
      {
        "rewardType": "Game",
        "gameDefinitionName": "SpintheWheelGame"
      }
    ]
  }
]
}

```

Here's a JSON example to create a transaction journal for a loyalty program member of an airline company.

```

{
  "transactionJournals": [
    {
      "ActivityDate": "2022-01-04T12:45:19Z",
      "JournalDate": "2022-01-04T00:45:19Z",
      "Brand": "Tata",
      "Establishment": "Vistara",
      "ExternalTransactionNumber": "P1-981950",
      "JournalTypeId": "01ET10000004CQp",
      "LoyaltyProgramId": "01pT10000004CdN",
      "MemberId": "01MT10000004CfA",
      "TransactionAmount": "117",
    }
  ]
}


```

```

    "TransactionLocation": "HitechCity",
    "VoucherCode": "",
    "Origin": "Bangalore",
    "Distance": "5",
    "Destination": "Hyderabad",
    "FlightNumber": "VS-Ran-001",
    "Status": "Pending"
  }
]
}

```

Using the transaction journal ID:

 **Note:** This is supported in API version 55.0 and later.

```

{
  "transactionJournals": [
    {
      "Id": "01Vxx00000000cj"
    }
  ]
}

```

#### Properties

Name	Type	Description	Required or Optional	Available Version
processInput	Map<String, Object>	List of the Transaction Journal records to process and create.	Required	54.0

#### Response body for POST

[Loyalty Engine Realtime Output](#)

## Transaction Journals Simulation

Runs a simulation to determine how transaction journals are processed against their corresponding loyalty program processes. When transaction journals meet the criteria and conditions for a program process, actions that are set up in the process are triggered for the transaction journals.

#### Special Access Rules

To use this resource, your org must have either B2C - Loyalty, B2C - Loyalty Plus, Loyalty Management - Growth, or Loyalty Management - Advanced license enabled, and you must be assigned the Loyalty Management permission set.

#### Resource

```
/connect/realtime/loyalty/programs/${programName}
```

#### Resource example

```

https://yourInstance.salesforce.com/services/data/vXX.X
/connect/realtime/loyalty/programs/StarAir_Program

```



**Available version**

55.0

**Requires Chatter**

No

**HTTP methods**

POST

**Request body for POST****Root XML tag**

&lt;loyaltyEngineInput&gt;

**JSON example**

 **Note:** Some of the API request fields contain field names that are interchangeable. For information on the list of interchangeable field names, see [Loyalty Management Connect API Interchangeable Request Fields](#).

```
{
  "shouldCheckCouponUsageLimit": true
  "transactionJournals": [
    {
      "ActivityDate": "2022-01-04T12:45:19Z",
      "JournalDate": "2022-01-04T00:45:19Z",
      "Brand": "",
      "Establishment": "Apple",
      "ExternalTransactionNumber": "P1-981950",
      "JournalTypeId": "01ET10000004CQp",
      "LoyaltyProgramId": "01pT10000004CdN",
      "MemberId": "01MT10000004CfA",
      "TransactionAmount": "117",
      "TransactionLocation": "HitechCity",
      "VoucherCode": "",
      "Origin": "asdf",
      "Distance": "5",
      "Destination": "Hyderabad",
      "FlightNumber": "SF-Ran-001",
      "Status": "Pending"
    }
  ],
  "runSetting": {
    "isSimulation": true
  }
}
```

 **Note:** Provide contactId to retrieve rewards related to a Person Account.

Using the transaction journal ID:

```
{
  "transactionJournals": [
    {
      "Id": "01Vxx00000000cj"
    }
  ],
  "runSetting": {
```

```

    "isSimulation": true
  }
}

```

Sample payload to retrieve rewards assigned to accounts.

```

{
  "transactionJournals": [
    {
      "ActivityDate": "2024-11-27T12:45:19Z",
      "JournalTypeId": "01ERM00000001Bb2AI",
      "LoyaltyProgramId": "01pRM00000002YTYAY",
      "ReferredPartyId": "001xx000003H3NnAAK",
      "Status": "Pending",
      "ProductId": "01txx0000006i44AAA",
      "ProductCategoryId": "0ZGxx0000000001GAA",
      "Quantity": "1",
      "TransactionAmount": "42",
      "appliedPromotions": [
        {
          "promotionId": "0c8xx00000000mPAAQ",
          "rewards": [
            {
              "rewardType": "Discount",
              "discountAmount": 5
            },
            {
              "rewardType": "Discount",
              "discountAmount": 10
            },
            {
              "rewardType": "Voucher",
              "voucherDefinitionName": "Voucher2",
              "voucherExpirationDate": "2025-08-07",
              "voucherEffectiveDate": "2024-08-08",
            },
            {
              "rewardType": "Game",
              "gameDefinitionName": "TestPostFacto",
            }
          ]
        }
      ]
    }
  ]
}

```

### Properties

Name	Type	Description	Required or Optional	Available Version
IsCouponUsageLimit	Boolean	Indicates whether the coupon usage limit check is enabled.	Required	62.0

Name	Type	Description	Required or Optional	Available Version
transactionJournals	<a href="#">Loyalty Engine Input Map</a>	Represents the transaction journals records to process.	Required	54.0
runSetting	<a href="#">Loyalty Engine Run Settings Input</a>	Loyalty runtime setting to run the simulation on the transaction journals.	Required for simulation, otherwise optional	55.0

**Response body for POST**[Loyalty Engine Realtime Output](#)**JSON example**

```
{
  "message": null,
  "processResult": null,
  "programName": "StarAir_Program",
  "simulationResults": {
    "transactionJournals": [
      {
        "executionSummary": {
          "pointsSummary": [
            {
              "changeInEscrowPointsBalance": 0,
              "changeInPointsBalance": 250,
              "loyaltyProgramCurrencyName": "VroomKing",
              "loyaltyProgramCurrencyType": "NonQualifying"
            },
            {
              "changeInEscrowPointsBalance": 20,
              "changeInPointsBalance": 0,
              "loyaltyProgramCurrencyName": "EcoMiles",
              "loyaltyProgramCurrencyType": "Qualifying"
            }
          ]
        },
        "vouchersIssuedCount": 0
      },
      {
        "processId": "9MXxx0000004Khg",
        "processName": "Redemption Process",
        "processRules": [
          {
            "actionResultSummary": [
              {
                "actionType": "CreditPoints",
                "changeInEscrowPointsBalance": 20,
                "changeInPointsBalance": 0,
                "escrowPointsCreditDate": "2022-03-05T00:00:00.000Z",
                "loyaltyProgramCurrencyName": "EcoMiles",
                "promotionId": "",
                "stepName": "Credit Brownie Points"
              },
              {

```

```

        "actionType": "CreditPoints",
        "changeInEscrowPointsBalance": 0,
        "changeInPointsBalance": 250,
        "loyaltyProgramCurrencyName": "VroomKing",
        "promotionId": "",
        "stepName": "Credit Base Points For Transaction"
    }
],
"executionStatus": "Success",
"executionSummary": {
    "pointsSummary": [
        {
            "changeInEscrowPointsBalance": 0,
            "changeInPointsBalance": 250,
            "loyaltyProgramCurrencyName": "VroomKing",
            "loyaltyProgramCurrencyType": "NonQualifying"
        },
        {
            "changeInEscrowPointsBalance": 20,
            "changeInPointsBalance": 0,
            "loyaltyProgramCurrencyName": "EcoMiles",
            "loyaltyProgramCurrencyType": "Qualifying"
        }
    ],
    "vouchersIssuedCount": 0
},
"processRuleName": "Rule 1",
"ruleSteps": [
    {
        "childSteps": [
            {
                "actionParameters": [
                    {
                        "name": "Notes",
                        "operator": "Equals",
                        "value": {
                            "expression": "Credit Brownie Points",
                            "expressionValue": {
                                "dataType": "Text",
                                "value": "Credit Brownie Points"
                            },
                            "parameterValues": {}
                        },
                    }
                ],
            },
            {
                "name": "ProgramCurrencyName",
                "operator": "Equals",
                "value": {
                    "expression": "EcoMiles",
                    "expressionValue": {
                        "dataType": "Text",
                        "value": "EcoMiles"
                    },
                    "parameterValues": {}
                },
            },
        ]
    }
]

```

```

        }
      },
      {
        "name": "PointsToCredit",
        "operator": "Equals",
        "value": {
          "expression": "20",
          "expressionValue": {
            "dataType": "Number",
            "value": 20
          },
          "parameterValues": {}
        }
      }
    ],
    "actionType": "CreditPoints",
    "sequenceNumber": 1,
    "status": "Success",
    "stepName": "Credit Brownie Points",
    "stepType": "Action"
  }
],
"filterCondition": "1",
"filterCriteria": [
  {
    "filterCriteriaResult": true,
    "operator": "Equals",
    "sequence": 1,
    "sourceFieldName": {
      "expression": "{!TransactionJournal.Product.Name}",
      "parameterValues": {
        "TransactionJournal.Product.Name": {
          "dataType": "Text",
          "value": "Brownie"
        }
      }
    }
  },
  {
    "value": {
      "expression": "Brownie",
      "expressionValue": {
        "dataType": "Text",
        "value": "Brownie"
      },
      "parameterValues": {}
    },
    "valueType": "FixedValue"
  }
],
"isConditionMet": true,
"sequenceNumber": 1,
"status": "Success",
"stepName": "Check Member Tier",
"stepType": "Condition"
},

```

```

    {
      "actionParameters": [
        {
          "name": "ProgramCurrencyName",
          "operator": "Equals",
          "value": {
            "expression": "VroomKing",
            "expressionValue": {
              "dataType": "Text",
              "value": "VroomKing"
            },
            "parameterValues": {}
          },
        },
        {
          "name": "PointsToCredit",
          "operator": "Equals",
          "value": {
            "expression": "{!DoublePoints}",
            "expressionValue": {
              "dataType": "Number",
              "value": 250
            },
            "parameterValues": {
              "TransactionJournal.TransactionAmount": {
                "dataType": "Number",
                "value": 125
              }
            },
            "resolvedExpression": "{(!TransactionJournal.TransactionAmount}
* 2)"
          },
        },
        {
          "name": "Notes",
          "operator": "Equals",
          "value": {
            "expression": "Base Points For Transaction",
            "expressionValue": {
              "dataType": "Text",
              "value": "Base Points For Transaction"
            },
            "parameterValues": {}
          },
        },
      ],
      "actionType": "CreditPoints",
      "sequenceNumber": 2,
      "status": "Success",
      "stepName": "Credit Base Points For Transaction",
      "stepType": "Action"
    },
  ]
}

```

```

    {
      "actionResultSummary": [],
      "errorMessage": "This rule didn't process the transaction journal because
the activity date of the transaction journal isn't between the start date and the end
date of the Holiday Season Promotion promotion.",
      "executionStatus": "Failed",
      "executionSummary": {
        "pointsSummary": [],
        "vouchersIssuedCount": 0
      },
      "processRuleName": "Rule 2",
      "promotionName": "Holiday Season Promotion",
      "ruleSteps": [
        {
          "actionParameters": [
            {
              "name": "VoucherExpirationDate",
              "operator": "Equals",
              "value": {
                "expression": "DATEVALUE(&quot;2023-01-31 00:00:00&quot;)",
                "expressionValue": {
                  "dataType": "Date",
                  "value": "DATEVALUE(&quot;2023-01-31 00:00:00&quot;)"
                },
              },
              "parameterValues": {}
            }
          ],
          "name": "VoucherDiscountPercentage",
          "operator": "Equals",
          "value": {
            "expression": "10",
            "expressionValue": {
              "dataType": "Number",
              "value": "10"
            },
            "parameterValues": {}
          }
        },
        {
          "name": "VoucherCode",
          "operator": "Equals",
          "value": {
            "expression": "&quot;QWDBHYWIGFOIHW&quot;",
            "expressionValue": {
              "dataType": "Text",
              "value": "&quot;QWDBHYWIGFOIHW&quot;"
            },
            "parameterValues": {}
          }
        },
        {
          "name": "VoucherEffectiveDate",
          "operator": "Equals",

```

```

        "value": {
          "expression": "DATEVALUE(&quot;2022-12-24 00:00:00&quot;)",
          "expressionValue": {
            "dataType": "Date",
            "value": "DATEVALUE(&quot;2022-12-24 00:00:00&quot;)"
          },
          "parameterValues": {}
        },
      },
    {
      "name": "Notes",
      "operator": "Equals",
      "value": {
        "expression": "Holiday Season Voucher",
        "expressionValue": {
          "dataType": "Text",
          "value": "Holiday Season Voucher"
        },
        "parameterValues": {}
      }
    },
    {
      "name": "VoucherDefinitionName",
      "operator": "Equals",
      "value": {
        "expression": "TimeBasedVoucherDef",
        "expressionValue": {
          "dataType": "Text",
          "value": "TimeBasedVoucherDef"
        },
        "parameterValues": {}
      }
    }
  ],
  "actionType": "IssueVoucher",
  "errorMessage": "",
  "sequenceNumber": 1,
  "status": "CriteriaNotMet",
  "stepName": "Issue Holiday Season Voucher",
  "stepType": "Action"
}
]
}
}
]
},
"status": true
}

```

## Loyalty Program Process

Run a loyalty process created from a loyalty program process template.



Out-of-the-box loyalty program process templates come with a default set of process parameters. You can configure new parameters or remove the default ones by simply cloning a template and running the configured parameters as a customized loyalty process using the loyalty program process API. For example, when using the *CreditPointsToMembers* process template to credit points to a loyalty member, you can configure the currency value so that the credited points can be redeemed in the currency you've specified.

The request and response parameters for the loyalty program process API vary depending on the input and output parameters configured for a loyalty program process in a Salesforce org.

### Resource

```
/connect/loyalty/programs/{programName}/program-processes/{processName}
```

### Resource examples

Loyalty Process API	Loyalty Process Template Name	Description
<a href="#">/connect/loyalty/programs/<i>DemoProgram</i>/program-processes/<i>CancelVoucher</i></a>	CancelVoucher	Cancel a voucher.
<a href="#">/connect/loyalty/programs/<i>DemoProgram</i>/program-processes/<i>CreditPointsToMembers</i></a>	CreditPointsToMembers	Credit pre-calculated points to a loyalty member on every transaction.
<a href="#">/connect/loyalty/programs/<i>DemoProgram</i>/program-processes/<i>DebitPointsFromMembers</i></a>	DebitPointsFromMembers	Redeem a loyalty member's pre-calculated points on a transaction.
<a href="#">/connect/loyalty/programs/<i>DemoProgram</i>/program-processes/<i>EnrollInPromotion</i></a>	EnrollInPromotion	Enroll a loyalty member to a promotion.
<a href="#">/connect/loyalty/programs/<i>DemoProgram</i>/program-processes/<i>GetMemberPromotions</i></a>	GetMemberPromotions	View eligible promotions of a loyalty member.
<a href="#">/connect/loyalty/programs/<i>DemoProgram</i>/program-processes/<i>IssueVoucher</i></a>	IssueVoucher	Issue a new voucher to a loyalty member.
<a href="#">/connect/loyalty/programs/<i>DemoProgram</i>/program-processes/<i>UnenrollMember</i></a>	UnenrollMember	Unenroll a member from a loyalty program.
<a href="#">/connect/loyalty/programs/<i>DemoProgram</i>/program-processes/<i>UpdateMemberDetails</i></a>	UpdateMemberDetails	Update a loyalty member profile.
<a href="#">/connect/loyalty/programs/<i>DemoProgram</i>/program-processes/<i>UpdateTier</i></a>	TierProcessing	Change the tier for a loyalty member.

### Available version

56.0

**Requires Chatter**

No

**HTTP methods**

POST

**Request parameters**

Name	Type	Description	Required or Optional	Available Version
programName	String	Name of the loyalty program.	Required	56.0
processName	String	Name of the process created from a loyalty program process template.	Required	56.0

**Request body for POST****Properties**

Name	Type	Description	Required or Optional	Available Version
processParameters	<a href="#">Loyalty Process Parameter Map</a>	Input parameters for running the loyalty process.	Required	56.0

**Response body for POST**[Loyalty Process Output](#)[Cancel a Voucher](#)

Change the status of a voucher to canceled.

[Credit Points to Members](#)

Credit pre-calculated points to a loyalty member on every transaction.

[Debit Points from Members](#)

Redeem a loyalty member's pre-calculated points on a transaction.

[Enroll for Promotions](#)

Enroll a loyalty member to a promotion.

[Get Member Promotions](#)

View all eligible promotions of a loyalty member.

[Issue a Voucher](#)

Issue a new voucher to a loyalty member.

[Unenroll a Member](#)

Unenroll a member from a loyalty program.

[Opt Out from a Promotion](#)

Opt out a loyalty program member from an available promotion.

[Update Member Details](#)

Update personal and preference details of a member.

[Update Member Tier](#)

Upgrade or downgrade tier for a loyalty member.

## Cancel a Voucher


Change the status of a voucher to canceled.

### Example URL

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/loyalty/programs/DemoProgram/program-processes/cancelVoucher
```

The request and response parameters vary depending on the input and output parameters configured for the *CancelVoucher* process type in a Salesforce org.

### Sample JSON Requests

 **Note:** Some of the API request fields contain field names that are interchangeable. For information on the list of interchangeable field names, see [Loyalty Management Connect API Interchangeable Request Fields](#).

Here's a sample request with a member ID and voucher ID

```
{
  "processParameters": [
    {
      "MemberId": "01Mxx00000000BJEAY",
      "VoucherId": "0kdxx000000000B245"
    }
  ]
}
```

Here's a sample request with a membership number and voucher code

 **Note:** If the voucher code isn't unique, provide either a voucher ID or a number.

```
{
  "processParameters": [
    {
      "MembershipNumber": "003",
      "VoucherCode": "15OFF"
    }
  ]
}
```

### Sample JSON Response

```
{
  "message": null,
  "outputParameters": {
    "outputParameters": {
```

```

    "results": [
      {
        "VoucherId": "0kDRM0000004Ct22AE"
      }
    ]
  },
  "simulationDetails": {
  },
  "status": true
}

```

SEE ALSO:

[Loyalty Management Help: Create a Process Using Cancel Voucher Process Template](#)

## Credit Points to Members

Credit pre-calculated points to a loyalty member on every transaction.

### Example URL


```

https://yourInstance.salesforce.com/services/data/v63.0/connect/loyalty/programs/DemoProgram/program-processes/credit
Points

```

The request and response parameters vary depending on the input and output parameters configured for the *CreditPointsToMembers* process type in a Salesforce org.

### Sample JSON Requests

 **Note:** Some of the API request fields contain field names that are interchangeable. For information on the list of interchangeable field names, see [Loyalty Management Connect API Interchangeable Request Fields](#).

Here's an example with a member ID:

```

{
  "processParameters": [
    {
      "TransactionJournal": {
        "ActivityDate": "2023-08-27T00:00:00Z",
        "ExternalTransactionNumber": "1",
        "InvoiceDate": "2023-08-23",
        "JournalTypeName": "Accrual",
        "MemberId": "01MRM0000004G5h2AE",
        "TransactionAmount": "45.0",
        "CurrencyIsoCode": "USD",
        "Status": "Processed"
      },
      "PointsToCredit": 100.0
    }
  ]
}

```

Here's an example with a membership number:

```
{
  "processParameters": [
    {
      "TransactionJournal": {
        "ActivityDate": "2023-08-27T00:00:00Z",
        "ExternalTransactionNumber": "1",
        "InvoiceDate": "2023-08-23",
        "JournalTypeName": "Accrual",
        "MembershipNumber": "M1256",
        "TransactionAmount": "45.0",
        "CurrencyIsoCode": "USD",
        "Status": "Processed"
      },
      "PointsToCredit": 100.0
    }
  ]
}
```

### Sample JSON Response

```
{
  "message": null,
  "outputParameters": {
    "outputParameters": {
      "results": [
        {
          "PointsCredited": 100.0
        }
      ]
    }
  },
  "simulationDetails": {
  },
  "status": true
}
```

SEE ALSO:

[Loyalty Management Help: Create a Process Using Credit Points to Member Process Template](#)

### Debit Points from Members


Redeem a loyalty member's pre-calculated points on a transaction.

### Example URL

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/loyalty/programs/DemoProgram/program-processes/debit
Points
```

The request and response parameters vary depending on the input and output parameters configured for the *DebitPointsFromMembers* process type in a Salesforce org.

## Sample JSON Requests

 **Note:** Some of the API request fields contain field names that are interchangeable. For information on the list of interchangeable field names, see [Loyalty Management Connect API Interchangeable Request Fields](#).

Here's an example with a member ID:

```
{
  "processParameters": [
    {
      "TransactionJournal": {
        "ActivityDate": "2023-08-27T00:00:00Z",
        "ExternalTransactionNumber": "1",
        "InvoiceDate": "2023-08-23",
        "JournalTypeName": "Redemption",
        "MemberId": "01MRM000004G5h2AE",
        "TransactionAmount": "45.0",
        "CurrencyIsoCode": "USD",
        "Status": "Processed"
      },
      "PointsToDebit": 200.0
    }
  ]
}
```

Here's an example with a membership number:

```
{
  "processParameters": [
    {
      "TransactionJournal": {
        "ActivityDate": "2023-08-27T00:00:00Z",
        "ExternalTransactionNumber": "1",
        "InvoiceDate": "2023-08-23",
        "JournalTypeName": "Redemption",
        "MembershipNumber": "M1256",
        "TransactionAmount": "45.0",
        "CurrencyIsoCode": "USD",
        "Status": "Processed"
      },
      "PointsToDebit": 200.0
    }
  ]
}
```

## Sample JSON Response

```
{
  "message": null,
  "outputParameters": {
    "outputParameters": {
```

```

      "results": [
        {
          "PointsDebited": 200.0
        }
      ]
    },
    "simulationDetails": {
    },
    "status": true
  }
}

```

SEE ALSO:

[Loyalty Management Help: Create a Process Using Debit Points from Member Process Template](#)

## Enroll for Promotions

Enroll a loyalty member to a promotion.

### Example URL

```

https://yourInstance.salesforce.com/services/data/v63.0/connect/loyalty/programs/DemoProgram/program-processes/EnrollPromotion

```

The request and response parameters vary depending on the input and output parameters configured for the *EnrollInPromotion* process type in a Salesforce org.

### Sample JSON Requests



**Note:** Some of the API request fields contain field names that are interchangeable. For information on the list of interchangeable field names, see [Loyalty Management Connect API Interchangeable Request Fields](#).

Here's an example with a member ID and promotion ID:

```

{
  "processParameters": [
    {
      "MemberId": "01Mxx00000000BJEAY",
      "PromotionId": "0c8xx00000000BJEAY"
    }
  ]
}

```

Here's an example with a membership number and promotion name:

```

{
  "processParameters": [
    {
      "MembershipNumber": "M1234",
      "PromotionName": "Flat20"
    }
  ]
}

```

```

    ]
  }

```

## Sample JSON Response

```

{
  "message":null,
  "outputParameters":{
    "outputParameters":{
      "results":[
        {
          "MemberId":"01MRM0000004GBn2AM",
          "LoyaltyProgramMbrPromotionId":"11pRM0000004CEmYAM"
        }
      ]
    }
  },
  "simulationDetails":{
  },
  "status":true
}

```

### SEE ALSO:

[Loyalty Management Help: Create a Process Using Enroll in Promotion Process Template](#)

## Get Member Promotions

View all eligible promotions of a loyalty member.

### Example URL

```

https://yourInstance.salesforce.com/services/data/v63.0/connect/loyalty/programs/DemoProgram/program-processes/ViewPromotions

```

The request and response parameters vary depending on the input and output parameters configured for the *GetMemberPromotions* process type in a Salesforce org.

## Sample JSON Requests



**Note:** Some of the API request fields contain field names that are interchangeable. For information on the list of interchangeable field names, see [Loyalty Management Connect API Interchangeable Request Fields](#).

Here's an example with a member ID:

```

{
  "processParameters":[
    {
      "MemberId":"01MRM0000004G5h2AE"
    }
  ]
}

```



Here's an example with a membership number:

```
{
  "processParameters": [
    {
      "MembershipNumber": "M1234"
    }
  ]
}
```

### Sample JSON Response

```
{
  "message": null,
  "outputParameters": {
    "outputParameters": {
      "results": [
        {
          "promotionEnrollmentRqr": false,
          "loyaltyPromotionType": "STANDARD",
          "memberEligibilityCategory": "Eligible",
          "promotionName": "doc_promo_1",
          "promotionId": "0c8RM0000004DnXYAU",
          "startDate": "2022-09-01"
        }
      ]
    }
  },
  "simulationDetails": {
  },
  "status": true
}
```

SEE ALSO:

[Loyalty Management Help: Create a Process Using Get Member Promotions Process Template](#)

### Issue a Voucher


Issue a new voucher to a loyalty member.

### Example URL

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/loyalty/programs/DemoProgram/program-processes/IssueVoucher
```

The request and response parameters vary depending on the input and output parameters configured for the *IssueVoucher* process type in a Salesforce org.

## Sample JSON Request

 **Note:** Some of the API request fields contain field names that are interchangeable. For information on the list of interchangeable field names, see [Loyalty Management Connect API Interchangeable Request Fields](#).

Here's an example with a member ID:

```
{
  "processParameters": [
    {
      "MemberId": "01Mxx00000000BJEAY",
      "VoucherCode": "15OFF",
      "VoucherFaceValue": 15,
      "VoucherExpirationDate": "2022-07-13"
    }
  ]
}
```

Here's an example with a membership number:

```
{
  "processParameters": [
    {
      "MembershipNumber": "003",
      "VoucherCode": "15OFF",
      "VoucherDiscountPercentage": 20
    }
  ]
}
```

## Sample JSON Response

```
{
  "message": null,
  "outputParameters": {
    "outputParameters": {
      "results": [
        {
        }
      ]
    }
  },
  "simulationDetails": {
  },
  "status": true
}
```

SEE ALSO:

[Loyalty Management Help: Create a Process Using Issue Voucher Process Template](#)

## Unenroll a Member


Unenroll a member from a loyalty program.

### Example URL

```
https://yourInstance.salesforce.com/services/data/v63.0/connect/loyalty/programs/DemoProgram/program-processes/unenrollMember
```

The request and response parameters vary depending on the input and output parameters configured for the *UnenrollMember* process type in a Salesforce org.

### Sample JSON Requests

 **Note:** Some of the API request fields contain field names that are interchangeable. For information on the list of interchangeable field names, see [Loyalty Management Connect API Interchangeable Request Fields](#).

Here's an example with a member ID:

```
{
  "processParameters": [
    {
      "MemberId": "01MRM0000004G6f2AE"
    }
  ]
}
```

Here's an example with a membership number:

```
{
  "processParameters": [
    {
      "CurrentMembershipNumber": "007"
    }
  ]
}
```

### Sample JSON Response

```
{
  "message": null,
  "outputParameters": {
    "outputParameters": {
      "results": [
        {
          "MemberId": "01MRM0000004F152AE"
        }
      ]
    }
  },
  "simulationDetails": {
  },
}
```

```

    "status":true
  }

```

SEE ALSO:

[Loyalty Management Help: Create a Process Using Unenroll Member Process Template](#)

## Opt Out from a Promotion

Opt out a loyalty program member from an available promotion.

### Example URL

```

https://yourInstance.salesforce.com/services/data/v63.0/connect/loyalty/programs/DemoProgram/program-processes/LeavePromotion

```

The request and response parameters vary depending on the input and output parameters configured for the *OptOutPromotion* process type in a Salesforce org.

### Sample JSON Requests



**Note:** Some of the API request fields contain field names that are interchangeable. For information on the list of interchangeable field names, see [Loyalty Management Connect API Interchangeable Request Fields](#).

Here's an example with a membership number and promotion name:

```

{
  "processParameters": [
    {
      "MembershipNumber": "M1",
      "PromotionName": "Promo1"
    }
  ]
}

```

Here's an example with a member ID and promotion ID:

```

{
  "processParameters": [
    {
      "MemberId": "01MRM0000004F152AE",
      "PromotionId": "11pxx00000000CvAAI"
    }
  ]
}

```

### Sample JSON Response

```

{
  "message": null,
  "outputParameters": {
    "outputParameters": {
      "results": [

```

```

        {
            "LoyaltyProgramMbrPromotionId": "1lpxx00000000CvAAI"
        }
    ]
}
},
"simulationDetails":{
},
"status":true
}

```

## Update Member Details

Update personal and preference details of a member.

### Example URL


```

https://yourInstance.salesforce.com/services/data/v63.0/connect/loyalty/programs/DemoProgram/program-processes/Update
Member Details

```

The request and response parameters vary depending on the input and output parameters configured for the *UpdateMemberDetails* process type in a Salesforce org.

### Sample JSON Request

 **Note:** Some of the API request fields contain field names that are interchangeable. For information on the list of interchangeable field names, see [Loyalty Management Connect API Interchangeable Request Fields](#).

Here's an example with a member ID:

```

{
  "processParameters": [
    {
      "MemberId": "01MRM0000004G5h2AE",
      "MembershipNumber": "M1234",
      "Email": "adam.smith@nto.com",
      "EnrollmentDate": "2023-05-28",
      "LastName": "Smith"
    }
  ]
}

```

Here's an example with a current membership number:

```

{
  "processParameters": [
    {
      "CurrentMembershipNumber": "M1234",
      "MembershipNumber": "M1234",
      "Email": "adam.smith@nto.com",
      "EnrollmentDate": "2023-05-28",
      "LastName": "Smith"
    }
  ]
}

```

```

    ]
  }

```

## Sample JSON Response

```

{
  "message":null,
  "outputParameters":{
    "outputParameters":{
      "results":[
        {
          "MemberId":"01MRM0000004F102AE",
          "ContactId":"003RM0000088CugYAE",
          "TransactionJournalId":null
        }
      ]
    }
  },
  "simulationDetails":{
  },
  "status":true
}

```

### SEE ALSO:

[Loyalty Management Help: Create a Process Using the Update Member Details Process Template](#)

## Update Member Tier

Upgrade or downgrade tier for a loyalty member.

### Example URL


```

https://yourInstance.salesforce.com/services/data/v63.0/connect/loyalty/programs/DemoProgram/program-processes/updateTier

```

The request and response parameters vary depending on the input and output parameters configured for the *TierProcessing* process type in a Salesforce org.

## Sample JSON Requests

 **Note:** Some of the API request fields contain field names that are interchangeable. For information on the list of interchangeable field names, see [Loyalty Management Connect API Interchangeable Request Fields](#).

Here's an example with a member ID:

```

{
  "processParameters":[
    {
      "MemberId":"01MRM0000004GBs2AM",
      "NewTier":"Gold",
      "ReasonForChange":"update tier"
    }
  ]
}

```

```
    }
  ]
}
```

Here's an example with a membership number:

```
{
  "processParameters": [
    {
      "MembershipNumber": "M2_Chloe_Sun",
      "NewTier": "Gold",
      "ReasonForChange": "update tier"
    }
  ]
}
```

## Sample JSON Response

```
{
  "message": null,
  "outputParameters": {
    "results": [
      {
        "CurrentTier": "Gold",
        "TierExpirationDate": "2023-10-31",
        "PreviousTier": "Silver",
        "EffectiveDate": "2023-09-07"
      }
    ]
  },
  "simulationDetails": {
  },
  "status": true
}
```

SEE ALSO:

[Loyalty Management Help: Create a Process Using the Change Member Tier Process Template](#)

## Gamification

Who doesn't like to play a game? Engage your customers more by gamifying customer activities, such as placing an order, or events, such as birthdays and anniversaries.

Assign eligible games to a customer, let the customer play the game, and then win a reward on an external system using the gamification REST APIs.

### Games

Get a list of eligible, expired, and played games, such as Spin the Wheel, or Scratch card, for a loyalty program member, an account, or a contact.

[Game Reward](#)

Get the details of the reward that's provided to a participant after the participant plays a Spin the Wheel, or Scratch Card game.

## Games

Get a list of eligible, expired, and played games, such as Spin the Wheel, or Scratch card, for a loyalty program member, an account, or a contact.

### Resource

```
/game/participant/${participantId}/games
```

### Resource Example

```
https://yourInstance.salesforce.com/services/data/v63.0/game/participant/01MRM0000004FHX2A2/games
```

```
https://yourInstance.salesforce.com/services/data/v63.0/game/participant/01MRM0000004FHX2A2/games?definitionId=3Gxx0000004CaGM
```

```
https://yourInstance.salesforce.com/services/data/v63.0/game/participant/01MRM0000004FHX2A2/games?gameParticipantRewardId=3PRSB000000hjJ4AQ
```

### Available version

60.0

### HTTP methods

GET

### Query parameters for GET

Parameter Name	Type	Description	Required or Optional	Available Version
definitionId	String	ID of the game definition associated with the game whose details are to be fetched.	Optional	60.0
gameParticipantRewardId	String	ID of the participant's game reward associated with the game whose details are to be fetched.	Optional	60.0

### Response body for GET

[Games List](#)

## Game Reward

Get the details of the reward that's provided to a participant after the participant plays a Spin the Wheel, or Scratch Card game.

### Resource

```
/game/gameparticipantreward/${gameParticipantRewardId}/game-reward
```

### Resource Example

```
https://yourInstance.salesforce.com/services/data/v63.0/game/gameparticipantreward/3PRSB000000hjJ4AQ/game-reward
```

### Available version

60.0



**HTTP methods**

GET

**Response body for GET**[Game Result](#)

## Promotion

The promotion APIs allow you to set eligibility rules and limits for a promotion, choose a promotion template, and create a promotion. You can also get the details of a specific promotion, and view a list of eligible promotions for a customer's cart.

[Eligible Promotions](#)

Get eligible promotions and their associated rewards for a cart and its line items.

[Promotion Configuration](#)

View configuration of a promotion, including its definition, eligibility criteria, rules, limits, and related rule library version. The Promotion Configuration API enables developers and system integrators in your company to view the configuration of active promotions in your Salesforce org and export the configuration to external systems such as offline point-of-sale (POS) systems. The Promotion Configuration API is only available in orgs where Global Promotions Management is available.

[Clone Promotion](#)

Clone an existing promotion along with its associated records

[Coupon Usage Increase](#)

Increase the usage count of a list of coupons.

[Coupon Usage Decrease](#)

Decrease the usage count of a list of coupons.

## Eligible Promotions

Get eligible promotions and their associated rewards for a cart and its line items.



**Note:** Ensure that the Context Definitions and Context Rule Library settings are enabled in your Salesforce org.

**Resource**

```
/global-promotions-management/eligible-promotions
```

**Resource example**

```
https://yourInstance.salesforce.com/services/data/v63.0/global-promotions-management/eligible-promotions
```

**Available version**

60.0

**HTTP methods**

POST

## Query Parameters

Name	Type	Description	Required or Optional	Available Version
ruleLibraryApiName	String	Enter the API Name of an active Rule Library with the Usage Type as Global Promotion Management. This parameter allows the API to check the eligibility of the cart against the Promotions associated with the specified Rule Library.	Optional	62.0

 **Note:** This query parameter is required if there are multiple active rule libraries configured with the Usage Type as Global Promotion Management.

## Request body for POST

### Root XML tag

CartInputMapRepresentation

### JSON examples

Here's an example of the details of a customer's cart whose eligible promotions are to be fetched:

 **Note:** Provide `contactId` to retrieve campaigns related to a Person Account.

```
{
  "cart": {
    "cartDetails": [
      {
        "activityStartDate": "2023-06-14T12:45:19Z",
        "contactId/loyaltyProgramMemberId/accountId": "003xx000004WjzAAAS",
        "transactionAmount": 499,
        "currencyISOCode": "USD",
        "engagementChannelId": "0eFxx0000000008EAA",
        "sourceLocationId/sourceLocation": "0YQxx0000000001GAA",
        "cartLineDetails": [
          {
            "cartLineProduct": "Burger",
            "cartLineProductCode": "testProductCode",
            "cartLineProductStockKeepingUnit": "testSKU",
            "cartLineProductCatalog": "Food",
            "cartLineItemQuantity": 1,
            "cartLineItemAmount": 100,
            "cartLineProductCategory": "test"
          }
        ]
      }
    ]
  }
}
```

Here's an example of the details of a loyalty program member's cart whose eligible loyalty program promotions are to be fetched:

```
{
  "cart": {
    "cartDetails": [
      {
        "activityStartDate": "2023-06-14T12:45:19Z",
        "contactId/loyaltyProgramMemberId/accountId": "003xx000004WjzAAAS",
        "currencyISOCode": "INR",
        "transactionAmount": 499,
        "membershipNumber": "Member-005",
        "cartLineDetails": [
          {
            "cartLineProductId": "01ttx0000006iTsAAI",
            "cartLineProductCode": "AnyProductCode",
            "cartLineProductStockKeepingUnit": "AnySKU",
            "cartLineItemQuantity": 2,
            "cartLineItemAmount": 99,
            "cartLineProductCatalogId": "0ZSSB0000002d8X4AQ"
          },
          {
            "cartLineProduct": "Coke",
            "cartLineProductCode": "AnyProductCode",
            "cartLineProductStockKeepingUnit": "AnySKU",
            "cartLineItemQuantity": 10,
            "cartLineItemAmount": 400,
            "cartLineProductCatalog": "Beverages"
          }
        ]
      }
    ]
  }
}
```

Here's an example to fetch eligible loyalty program promotions for accounts:

```
{
  "cart": {
    "cartDetails": [
      {
        "activityStartDate": "2024-10-16T22:45:19Z",
        "accountId": "001xx000003H3NnAAK",
        "transactionAmount": 40,
        "currencyISOCode": "USD",
        "cartLineDetails": [
          {
            "cartLineProductCatalog": "Beverages",
            "cartLineProductCategory": "Coffee Creations",
            "cartLineProduct": "Caramel Latte",
            "cartLineItemQuantity": 5
          }
        ]
      }
    ]
  }
}
```

```

    }
  }
}

```

### Properties

Name	Type	Description	Required or Optional	Available Version
cart	Map<String, Object>	<p>Fields configured in the <b>Cart</b> and <b>CartLine</b> nodes of the <i>CartDetailGetEligiblePromotion</i> context service definition in your org. These fields represent the details of a cart and its line items whose eligible promotions are to be fetched.</p> <p>To get eligible promotions associated with a loyalty program member's cart, the required parameters are:</p> <ul style="list-style-type: none"> <li>• membershipNumber Or loyaltyProgramMemberId</li> <li>• currencyISOCode</li> </ul> <p>For other eligible promotions, the required parameters are:</p> <ul style="list-style-type: none"> <li>• contactId, contactLastName, or contactEmail</li> <li>• currencyISOCode</li> </ul> <p>Along with the required parameters, provide specific details of a cart, such as the total transaction amount, and the cart line item details, such as the line item quantity and amount, the product ID or name, the product catalog ID or name, the product code, and the product stock keeping unit.</p>	Required	60.0

### Response body for POST

[Eligible Promotions List](#)

## Promotion Configuration

View configuration of a promotion, including its definition, eligibility criteria, rules, limits, and related rule library version. The Promotion Configuration API enables developers and system integrators in your company to view the configuration of active promotions in your Salesforce org and export the configuration to external systems such as offline point-of-sale (POS) systems. The Promotion Configuration API is only available in orgs where Global Promotions Management is available.

**Resource**

```
/global-promotions-management/promotions/${promotionId}/configuration
```

**Available version**

61.0

**HTTP methods**

GET

**Request parameters for GET**

Parameter Name	Type	Description	Required or Optional	Available Version
promotionId	String	ID of the active promotion you want to view. The ID must belong to a promotion whose rules are configured using a non-Loyalty promotion template.	Required	61.0
relatedObjectFieldNames	String[]	The list of related object fields whose details you want to view for the promotion. To view the details of a field, specify the API name of the object that the field belongs to and the API name of the field, separated by a period.	Optional	61.0

**Response body for GET**

[Promotion Configuration Output](#) on page 425

**Clone Promotion**

Clone an existing promotion along with its associated records

**Resource**

```
/promotions/clone
```

**Available version**

61.0

**HTTP methods**

POST

**JSON example**

```
{
  "sourcePromotionId": "0c8xx00000000WHAAY",
  "name": "ups clone 4",
  "startDateTime": "2024-01-08T00:00:00.000Z",
  "endDateTime": "2024-01-09T00:00:00.000Z",
  "promotionCode": "sdfgy543wsd",
  "doesReuseEngagementAttributes": "true",
  "additionalFieldValues": {
    "additionalFieldValuesMap": {
```

```

    "Description": "test ups clone"
  }
},
"relatedRecords": {
  "relatedRecordsList": [
    {
      "relatedObjectApiName": "PromotionChannel",
      "relatedObjectPromotionFieldName": "PromotionId"
    },
    {
      "relatedObjectApiName": "PromotionLoyaltyPtrnrProdt",
      "relatedObjectPromotionFieldName": "PromotionId"
    },
    {
      "relatedObjectApiName": "PromotionMarketSegment",
      "relatedObjectPromotionFieldName": "PromotionId"
    },
    {
      "relatedObjectApiName": "PromotionProduct",
      "relatedObjectPromotionFieldName": "PromotionId"
    },
    {
      "relatedObjectApiName": "PromotionProductCategory",
      "relatedObjectPromotionFieldName": "PromotionId"
    },
    {
      "relatedObjectApiName": "PromotionLimit",
      "relatedObjectPromotionFieldName": "PromotionId"
    }
  ]
}
}

```

### Properties

Name	Type	Description	Required or Optional	Available Version
additionalFieldValue	Map<String, String>	The details of the promotion's additional fields.	Small, 61.0	61.0
useEngageAttributes	Boolean	Indicates whether the new promotion uses the engagement attributes created for the source promotion to track customers' promotion usage.	Small, 61.0	61.0
endDateTime	String	The end date and time of the new promotion.	Small, 61.0	61.0
name	String	The name of the new promotion.	Small, 61.0	61.0
promotionCode	String	The code of the new promotion.	Small, 61.0	61.0

Name	Type	Description	Required or Optional	Available Version
relatedRecords	<a href="#">Related Promotion Records Input []</a> on page 372	The related record for the source promotion that have to be created for the new promotion.	Small, 61.0	61.0
sourcePromotionId	String	The ID of the promotion that's cloned.	Small, 61.0	61.0
startDateTime	String	The start date and time of the new promotion.	Small, 61.0	61.0

**Response body for POST**

[Clone Promotion Output](#) on page 440

**Coupon Usage Increase**

Increase the usage count of a list of coupons.



**Example:**

**Resource**

```
/services/data/vXX.X/global-promotions-management/coupons/usage-increase
```

**Examples**

```
https://yourInstance.salesforce.com/services/data/vXX.X  
/global-promotions-management/coupons/usage-increase
```

```
https://yourInstance.salesforce.com/services/data/vXX.X  
/global-promotions-management/coupons/usage-increase
```

**Available version**

62.0

**Requires Chatter**

No

**HTTP methods**

POST

**Request body for POST**

The request body is available in version 62.0 and later.

**JSON example**

```
{
  "couponCodeList": [
    "QWERTY",
    "ABCD"
  ],
  "activityDateTime": "2024-10-05T12:45:19Z",
  "externalTransactionNumber": "01Vxx000000006T",
  "customerId": "01Mxx0000000001"
}
```

**Properties**

Name	Type	Description	Required or Optional	Available Version
couponCodeList	List<String>	The codes of the list of coupons for which the usage must be increased or decreased.	Required	62.0
activityDateTime	DateTime	The date and time at which the customer redeemed the coupon.	Optional	62.0
externalTransactionId	String	The externally generated unique identifier of cart for which the customer redeemed the coupon.	Optional	62.0
customerId	String	The ID of either the contact, the account, or the loyalty program member who redeemed the coupon.	Optional	62.0

**Response body for POST**

[Coupon Usage Result](#) on page 441

## Coupon Usage Decrease

Decrease the usage count of a list of coupons.

 **Example:**

**Resource**

```
/services/data/vXX.X/global-promotions-management/coupons/usage-decrease
```

**Examples**

```
https://yourInstance.salesforce.com/services/data/vXX.X  
/global-promotions-management/coupons/usage-decrease
```

```
https://yourInstance.salesforce.com/services/data/vXX.X  
/global-promotions-management/coupons/usage-decrease
```

**Available version**

62.0

**Requires Chatter**

No

**HTTP methods**

POST

**Request body for POST**

The request body is available in version 62.0 and later.



**JSON example**

```
{
  "couponCodeList": [
    "QWERTY",
    "ABCD"
  ],
  "activityDateTime": "2024-10-05T12:45:19Z",
  "externalTransactionNumber": "01Vxx000000006T",
  "customerId": "01Mxx0000000001"
}
```

**Properties**

Name	Type	Description	Required or Optional	Available Version
couponCodeList	List<String>	The codes of the list of coupons for which the usage must be increased or decreased.	Required	62.0
activityDateTime	DateTime	The date and time at which the customer redeemed the coupon.	Optional	62.0
externalTransactionNumber	String	The externally generated unique identifier of cart for which the customer redeemed the coupon.	Optional	62.0
customerId	String	The ID of either the contact, the account, or the loyalty program member who redeemed the coupon.	Optional	62.0

**Response body for POST**

[Coupon Usage Result](#) on page 441

## Request Bodies

Here's a list of Loyalty Management API request bodies.

[Cart Input Map](#)

Input representation of the summary of the cart whose eligible promotions are to be fetched.

[Corporate Member Enrollment Input](#)

Input representation for creating a business account and a Corporate type Loyalty Program Member record for the member.

[Individual Member Enrollment Input](#)

Input representation for creating a person account and an Individual type Loyalty Program Member record for the member.

[Loyalty Engine Input](#)

Input representation of the Loyalty Engine Runtime API.

[Loyalty Engine Input Map](#)

Input representation of the Transaction Journal record execution request.

[Loyalty Engine Run Settings Input](#)

Input representation of the loyalty engine settings to run simulation on the transaction journals.

[Loyalty Account Input](#)

Input representation for creating a business account for the member.

[Loyalty Contact Input](#)

Input representation of the loyalty program member's person account.

[Loyalty Process Input](#)

Input representation of the details of the loyalty program process action.

[Loyalty Process Input Options](#)

Input representation of the execution options for the loyalty process.

[Loyalty Process Parameter Map Input](#)

Input representation of the input parameters for the loyalty process.

[Loyalty View Transaction Journal History Input](#)

Input representation for getting the transaction journal records of a loyalty member.

[Redeem Voucher Input](#)

Input representation for voucher redemption.

[Clone Promotion Input](#)

Input representation that decides which existing promotion is cloned and the attributes of the new promotion that's created.

[Related Promotion Records Input](#)

Input representation that decides which related records and fields of the existing promotion are cloned and associated with the new promotion.

[Coupon Usage Input](#)

Input representation of Coupon Usage.

## Cart Input Map

Input representation of the summary of the cart whose eligible promotions are to be fetched.

### Root XML tag

`CartInputMapRepresentation`

### JSON examples

Here's an example of the details of a customer's cart whose eligible promotions are to be fetched:

 **Note:** Provide `contactId` to retrieve campaigns related to a Person Account.

```

{
  "cart": {
    "cartDetails": [
      {
        "activityStartDate": "2023-06-14T12:45:19Z",
        "contactId/loyaltyProgramMemberId/accountId": "003xx000004WjzAAAS",
        "transactionAmount": 499,
        "currencyISOCode": "USD",
      }
    ]
  }
}

```

```

    "engagementChannelId": "0eFxx0000000008EAA",
    "sourceLocationId/sourceLocation": "0YQxx0000000001GAA",
    "cartLineDetails": [
      {
        "cartLineProduct": "Burger",
        "cartLineProductCode": "testProductCode",
        "cartLineProductStockKeepingUnit": "testSKU",
        "cartLineProductCatalog": "Food",
        "cartLineItemQuantity": 1,
        "cartLineItemAmount": 100,
        "cartLineProductCategory": "test"
      }
    ]
  }
}

```

Here's an example of the details of a loyalty program member's cart whose eligible loyalty program promotions are to be fetched:

```

{
  "cart": {
    "cartDetails": [
      {
        "activityStartDate": "2023-06-14T12:45:19Z",
        "contactId/loyaltyProgramMemberId/accountId": "003xx000004WjzAAAS"
        "currencyISOCODE": "INR",
        "transactionAmount": 499,
        "membershipNumber": "Member-005",
        "cartLineDetails": [
          {
            "cartLineProductId": "01txx0000006iTsAAI",
            "cartLineProductCode": "AnyProductCode",
            "cartLineProductStockKeepingUnit": "AnySKU",
            "cartLineItemQuantity": 2,
            "cartLineItemAmount": 99,
            "cartLineProductCatalogId": "0ZSSB0000002d8X4AQ"
          },
          {
            "cartLineProduct": "Coke",
            "cartLineProductCode": "AnyProductCode",
            "cartLineProductStockKeepingUnit": "AnySKU",
            "cartLineItemQuantity": 10,
            "cartLineItemAmount": 400,
            "cartLineProductCatalog": "Beverages"
          }
        ]
      }
    ]
  }
}

```

Here's an example to fetch eligible loyalty program promotions for accounts:

```
{
  "cart": {
    "cartDetails": [
      {
        "activityStartDate": "2024-10-16T22:45:19Z",
        "accountId": "001xx000003H3NnAAK",
        "transactionAmount": 40,
        "currencyISOCODE": "USD",
        "cartLineDetails": [
          {
            "cartLineProductCatalog": "Beverages",
            "cartLineProductCategory": "Coffee Creations",
            "cartLineProduct": "Caramel Latte",
            "cartLineItemQuantity": 5
          }
        ]
      }
    ]
  }
}
```

### Properties

Name	Type	Description	Required or Optional	Available Version
cart	Map<String, Object>	<p>Fields configured in the <b>Cart</b> and <b>CartLine</b> nodes of the <i>CartDetailGetEligiblePromotion</i> context service definition in your org. These fields represent the details of a cart and its line items whose eligible promotions are to be fetched.</p> <p>To get eligible promotions associated with a loyalty program member's cart, the required parameters are:</p> <ul style="list-style-type: none"> <li>• membershipNumber or loyaltyProgramMemberId</li> <li>• currencyISOCODE</li> </ul> <p>For other eligible promotions, the required parameters are:</p> <ul style="list-style-type: none"> <li>• contactId, contactLastName, or contactEmail</li> <li>• currencyISOCODE</li> </ul> <p>Along with the required parameters, provide specific details of a cart, such as the total transaction amount, and the cart</p>	Required	60.0

Name	Type	Description	Required or Optional	Available Version
		line item details, such as the line item quantity and amount, the product ID or name, the product catalog ID or name, the product code, and the product stock keeping unit.		

## Corporate Member Enrollment Input

Input representation for creating a business account and a Corporate type Loyalty Program Member record for the member.

### Root XML tag

```
<corporateMemberEnrollment>
```

### JSON example

```
{
  "enrollmentDate" : "2021-01-01T05:05:05",
  "membershipNumber" : "Member123",
  "associatedAccountDetails" : {
    "name" : "Name1",
    "phone" : "908989898",
    "website" : "www.ab.com",
    "allowDuplicateRecords" : "false",
    "additionalAccountFieldValues" : {
      "attributes" : {
        "CustomNumber__c" : "20.0"
      }
    }
  },
  "memberStatus" : "Active",
  "referredBy" : "Member12",
  "referredByMemberReferralCode" : "MPX12adf",
  "createTransactionJournals" : "true",
  "transactionJournalStatementFrequency" : "Monthly",
  "additionalMemberFieldValues" : {
    "attributes" : {
      "CustomBoolean__c" : "true"
    }
  }
}
```

### Properties

Name	Type	Description	Required or Optional	Available Version
additionalMemberFieldValues	Map<String, String>	The custom attributes of the member in the loyalty program member record. <ul style="list-style-type: none"> <li>Keys</li> <li>Values</li> </ul>	Optional	55.0

Name	Type	Description	Required or Optional	Available Version
associateAccountDetails	Loyalty Account Input	The business account details of the member.	Required	55.0
canReceivePartner Promotions	Boolean	Indicates whether the member can receive partner promotions for the program ( <code>true</code> ) or not ( <code>false</code> ).	Optional	55.0
canReceivePromotions	Boolean	Indicates whether the member can receive promotions for the program.	Optional	55.0
createTransaction Journals	Boolean	Indicates whether to create the transaction journal records ( <code>true</code> ) or not ( <code>false</code> ): The transaction journal records are created to: <ul style="list-style-type: none"> <li>• Credit points to the member for enrolling.</li> <li>• Credit points to the member who has referred the new member.</li> </ul>	Optional	55.0
enrollmentChannel	String	The channel used by members to enroll into a loyalty program. Possible values are: <ul style="list-style-type: none"> <li>• CallCenter</li> <li>• Email</li> <li>• Franchise</li> <li>• Mobile</li> <li>• Partner</li> <li>• Pos</li> <li>• Print</li> <li>• Social</li> <li>• Store</li> <li>• Web</li> </ul>	Optional	55.0
enrollmentDate	String	The date on which a member is enrolled.	Required	55.0
memberStatus	String	Status of the member. Possible values are: <ul style="list-style-type: none"> <li>• Active</li> <li>• Inactive</li> <li>• Custom value</li> </ul>	Required	55.0
membershipEndDate	String	The date on which the loyalty program member's membership ends.	Optional	55.0

Name	Type	Description	Required or Optional	Available Version
membershipNumber	String	The unique membership number of the loyalty program member.	Required	55.0
referredBy	String	The ID of the loyalty program member who has referred the member.	Optional	55.0
<del>referredBy</del> referralCode	String	The referral code of the loyalty program member who has referred the member.	Optional	57.0
transactionJournal StatementFrequency	String	The frequency at which transaction journal statements must be delivered to the member. Possible values are: <ul style="list-style-type: none"> <li>• Daily</li> <li>• Monthly</li> <li>• Yearly</li> </ul>	Optional	55.0
transactionJournal StatementMethod	String	The method used to deliver transaction journal statements to the member. Possible values are: <ul style="list-style-type: none"> <li>• Email</li> <li>• Mail</li> </ul>	Optional	55.0

## Individual Member Enrollment Input

Input representation for creating a person account and an Individual type Loyalty Program Member record for the member.

### Root XML tag

```
<individualMemberEnrollment>
```

### JSON example

```
{
  "enrollmentDate" : "2021-01-01T00:00:00",
  "membershipNumber" : "Member1",
  "associatedContactDetails" : {
    "firstName" : "Name1",
    "lastName" : "Name2",
    "email" : "ab@email.com",
    "allowDuplicateRecords" : "false",
    "additionalContactFieldValues" : {
      "attributes" : {
        "PersonMailingCity" : "Delhi",
        "CustomNumber__c" : "20.0"
      }
    }
  },
  "memberStatus" : "Active",
  "referredBy" : "Member2",
```

```

"referredByMemberReferralCode" : "MPX12adf",
"createTransactionJournals" : "true",
"transactionJournalStatementFrequency" : "Monthly",
"transactionJournalStatementMethod" : "Mail",
"enrollmentChannel" : "Email",
"canReceivePromotions": "true",
"canReceivePartnerPromotions" : "true",
"membershipEndDate" : "2022-01-01T00:00:00",
"relatedCorporateMembershipNumber" : "Member3",
"additionalMemberFieldValues" : {
  "attributes" : {
    "CustomBoolean__c" : "true"  }
}
}

```

### Properties

Name	Type	Description	Required or Optional	Available Version
additionalMemberFieldValues	Map<String, String>	The custom attributes of the member in the loyalty program member record. <ul style="list-style-type: none"> <li>Keys</li> <li>Values</li> </ul>	Required	55.0
associatedContact on page 367Details	<a href="#">Loyalty Contact Input on page 367</a>	The person account record details of the individual type member.	Required	55.0
canReceivePartnerPromotions	Boolean	Indicates whether the member can receive partner promotions for the program ( <code>true</code> ) or not ( <code>false</code> ).	Optional	55.0
canReceivePromotions	Boolean	Indicates whether the loyalty program member can receive promotions.	Optional	55.0
createTransactionJournals	Boolean	Indicates whether to create the transaction journal records ( <code>true</code> ) or not ( <code>false</code> ): The transaction journal records are created to: <ul style="list-style-type: none"> <li>Credit points to the member for enrolling.</li> <li>Credit points to the member who has referred the new member.</li> </ul>	Optional	55.0
enrollmentChannel	String	The channel used to by loyalty program member to enroll in to the loyalty program. Possible values are: <ul style="list-style-type: none"> <li>Pos</li> <li>Web</li> </ul>	Optional	55.0



Name	Type	Description	Required or Optional	Available Version
		<ul style="list-style-type: none"> <li>Email</li> <li>CallCenter</li> <li>Social</li> <li>Mobile</li> <li>Store</li> <li>Franchise</li> <li>Partner</li> <li>Print</li> </ul>		
enrollmentDate	Date	<p>The date on which the loyalty program members are enrolled in to the loyalty program.</p> <p>The format is: 2020-09-08T00:00:00</p>	Required	55.0
memberStatus	String	<p>The status of the loyalty program member.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>Active</li> <li>Inactive</li> <li>Custom</li> </ul>	Required	55.0
membershipEndDate	String	The date on which the loyalty program member's membership ends.	Optional	55.0
membershipNumber	String	The unique membership number of the loyalty program member.	Required	55.0
referredBy	String	The membership number of the loyalty program member who has referred the member.	Optional	55.0
referredByMemberReferralCode	String	The referral code of the loyalty program member who has referred the member.	Optional	57.0
relatedCorporateMembershipNumber	String	External corporate membership number of the member.	Optional	55.0
transactionJournalStatementFrequency	String	<p>The frequency at which transaction journal statements must be delivered to the member. Possible values are:</p> <ul style="list-style-type: none"> <li>Monthly</li> <li>Quarterly</li> </ul>	Optional	55.0

Name	Type	Description	Required or Optional	Available Version
transactionJournalStatementMethod	String	The method used to deliver transaction journal statements to the member. Possible values are: <ul style="list-style-type: none"> <li>• Mail</li> <li>• Email</li> </ul>	Optional	55.0


## Loyalty Engine Input

Input representation of the Loyalty Engine Runtime API.

### Root XML tag

```
<loyaltyEngineInput>
```

### JSON example

 **Note:** Some of the API request fields contain field names that are interchangeable. For information on the list of interchangeable field names, see [Loyalty Management Connect API Interchangeable Request Fields](#).

```
{
  "shouldCheckCouponUsageLimit": true
  "transactionJournals": [
    {
      "ActivityDate": "2022-01-04T12:45:19Z",
      "JournalDate": "2022-01-04T00:45:19Z",
      "Brand": "",
      "Establishment": "Apple",
      "ExternalTransactionNumber": "P1-981950",
      "JournalTypeId": "01ET10000004CQp",
      "LoyaltyProgramId": "01pT10000004CdN",
      "MemberId": "01MT10000004CfA",
      "TransactionAmount": "117",
      "TransactionLocation": "HitechCity",
      "VoucherCode": "",
      "Origin": "asdf",
      "Distance": "5",
      "Destination": "Hyderabad",
      "FlightNumber": "SF-Ran-001",
      "Status": "Pending"
    }
  ],
  "runSetting": {
    "isSimulation": true
  }
}
```

 **Note:** Provide `contactId` to retrieve rewards related to a Person Account.

Using the transaction journal ID:

```
{
  "transactionJournals": [
    {
      "Id": "01Vxx00000000cj"
    }
  ],
  "runSetting": {
    "isSimulation": true
  }
}
```

Sample payload to retrieve rewards assigned to accounts.

```
{
  "transactionJournals": [
    {
      "ActivityDate": "2024-11-27T12:45:19Z",
      "JournalTypeId": "01ERM00000001Bb2AI",
      "LoyaltyProgramId": "01pRM000000002YTYAY",
      "ReferredPartyId": "001xx000003H3NnAAK",
      "Status": "Pending",
      "ProductId": "01txx0000006i44AAA",
      "ProductCategoryId": "0ZGxx0000000001GAA",
      "Quantity": "1",
      "TransactionAmount": "42",
      "appliedPromotions": [
        {
          "promotionId": "0c8xx00000000mPAAQ",
          "rewards": [
            {
              "rewardType": "Discount",
              "discountAmount": 5
            },
            {
              "rewardType": "Discount",
              "discountAmount": 10
            },
            {
              "rewardType": "Voucher",
              "voucherDefinitionName": "Voucher2",
              "voucherExpirationDate": "2025-08-07",
              "voucherEffectiveDate": "2024-08-08",
            },
            {
              "rewardType": "Game",
              "gameDefinitionName": "TestPostFacto",
            },
          ]
        }
      ]
    }
  ]
}
```

## Properties

Name	Type	Description	Required or Optional	Available Version
enableCouponUsageLimit	Boolean	Indicates whether the coupon usage limit check is enabled.	Required	62.0
transactionJournals	Loyalty Engine Input Map	Represents the transaction journals records to process.	Required	54.0
runSetting	Loyalty Engine Run Settings Input	Loyalty runtime setting to run the simulation on the transaction journals.	Required for simulation, otherwise optional	55.0


## Loyalty Engine Input Map

Input representation of the Transaction Journal record execution request.

### JSON example

In the JSON, you can either provide the details of transaction journals that aren't available in the org, or a list of IDs that belong to transaction journals available in the org. If you provide the details of transaction journals that aren't available in the org, each transaction journal's applicable loyalty program process applies the eligible rules, executes the applicable actions, and then creates the transaction journal.

Using the transaction journal details:

 **Note:** Some of the API request fields contain field names that are interchangeable. For information on the list of interchangeable field names, see [Loyalty Management Connect API Interchangeable Request Fields](#).

Here's a JSON example that creates a transaction journal for a purchase-related transaction made by a loyalty program member of a retail company. It has all the transaction journal-related information along with the details of the promotion applied, including the different types of rewards the transaction is eligible for. For information on the TransactionJournal object fields, see [https://developer.salesforce.com/docs/atlas.en-us.loyalty.meta/loyalty/sforce\\_api\\_objects\\_transactionjournal.htm](https://developer.salesforce.com/docs/atlas.en-us.loyalty.meta/loyalty/sforce_api_objects_transactionjournal.htm).

```
{
  "transactionJournals": [
    {
      "ActivityDate": "2023-11-27T12:45:19Z",
      "JournalTypeId": "01ERM0000001Bb2AI",
      "LoyaltyProgramId": "01pRM00000002YTYAY",
      "MemberId": "01MRM0000002Fmu2AE",
      "Status": "Pending",
      "ProductId": "01txx0000006iTsAAI",
      "ProductCategoryId": "0zSSB0000002d8X4AQ",
      "Quantity": "4",
      "TransactionAmount": "500",
      "appliedPromotions": [
        {
          "promotionId": "0c8RM0000004FiXYAU",
          "rewards": [
            {
              "rewardType": "Discount",
              "discountAmount": 15
            }
          ]
        }
      ]
    }
  ]
}
```

```

    },
    {
      "rewardType": "Points",
      "loyaltyProgramCurrencyName": "Coins",
      "points": 100,
      "relatedInformation": "Test",
      "notes": "points reward"
    },
    {
      "rewardType": "Voucher",
      "voucherDefinitionName": "10% Discount Voucher",
      "voucherExpirationDate": "2023-12-15",
      "voucherEffectiveDate": "2023-11-29",
      "notes": "discount voucher"
    },
    {
      "rewardType": "Badge",
      "loyaltyProgramBadgeName": "Gen AI Badge",
      "memberBadgeStatus": "Active",
      "badgeValidityEndDate": "2023-12-15",
      "reason": "purchase"
    },
    {
      "rewardType": "Game",
      "gameDefinitionName": "SpintheWheelGame"
    }
  ]
}
]
}
]
}
}

```

Here's a JSON example to create a transaction journal for a loyalty program member of an airline company.


```

{
  "transactionJournals": [
    {
      "ActivityDate": "2022-01-04T12:45:19Z",
      "JournalDate": "2022-01-04T00:45:19Z",
      "Brand": "Tata",
      "Establishment": "Vistara",
      "ExternalTransactionNumber": "P1-981950",
      "JournalTypeId": "01ET10000004CQp",
      "LoyaltyProgramId": "01pT10000004CdN",
      "MemberId": "01MT10000004CfA",
      "TransactionAmount": "117",
      "TransactionLocation": "HitechCity",
      "VoucherCode": "",
      "Origin": "Bangalore",
      "Distance": "5",
      "Destination": "Hyderabad",
      "FlightNumber": "VS-Ran-001",
      "Status": "Pending"
    }
  ]
}

```

```
]
}
```

Using the transaction journal ID:

 **Note:** This is supported in API version 55.0 and later.

```
{
  "transactionJournals": [
    {
      "Id": "01Vxx00000000cj"
    }
  ]
}
```

### Properties

Name	Type	Description	Required or Optional	Available Version
processInput	Map<String, Object>	List of the Transaction Journal records to process and create.	Required	54.0

## Loyalty Engine Run Settings Input

Input representation of the loyalty engine settings to run simulation on the transaction journals.

### Root XML tag

```
<loyaltyEngineRunSettingsRepresentation>
```

### JSON example

```
{
  "runSetting": {
    "isSimulation": true
  }
}
```

### Properties

Name	Type	Description	Required or Optional	Available Version
isSimulation	Boolean	Indicates whether to run a simulation on the transaction journals ( <code>true</code> ) or not ( <code>false</code> ). Default value is <code>false</code> .	Required for simulation otherwise optional.	55.0

## Loyalty Account Input

Input representation for creating a business account for the member.

### Root XML tag

```
<loyaltyAccountInput>
```

**JSON example**

```
{
  "associatedAccountDetails": {
    "name": "Name1",
    "phone": "908989898",
    "website": "www.ab.com",
    "allowDuplicateRecords": "false",
    "additionalAccountFieldValues": {
      "attributes": {
        "CustomNumber__c": "20.0"
      }
    }
  }
}
```

**Properties**

Name	Type	Description	Required or Optional	Available Version
additionalAccountFieldValues	Map<String, String>	The additional attribute values that must be saved in the account record.	Optional	55.0
allowDuplicateRecords	Boolean	Indicates whether a duplicate member record can be created, when an existing record with identical details exists in the org.	Optional	55.0
name	String	Name of the account.	Required	55.0
phone	String	The phone number associated with the account.	Optional	55.0
website	String	The website associated with the account.	Optional	55.0

**Loyalty Contact Input**

Input representation of the loyalty program member's person account.

**Root XML tag**

```
<loyaltyContactInput>
```

**JSON example**

```
{
  "enrollmentDate" : "2021-01-01T00:00:00",
  "membershipNumber" : "Member1",
  "associatedContactDetails" : {
    "firstName" : "Name1",
    "lastName" : "Name2",
    "email" : "ab@gmail.com",
    "allowDuplicateRecords" : "false",
    "additionalContactFieldValues" : {
      "attributes" : {
```

```

    "PersonMailingCity" : "Delhi",
    "CustomNumber__c" : "20.0"    }
}

```

### Properties

Name	Type	Description	Required or Optional	Available Version
additionalContactFieldValues	Map<String, String>	The additional attributes that must be saved in the person account.	Optional	55.0
allowDuplicateRecords	Boolean	Indicates if a duplicate person account record can be created for a member when an existing person account record exists.  The default value is <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">True</div>	Optional	55.0
email	String	Email ID of the person account.	Optional	55.0
firstName	String	First name of the person account.	Optional	55.0
lastName	String	Last name of the person account.	Required	55.0

## Loyalty Process Input

Input representation of the details of the loyalty program process action.

### Properties

Name	Type	Description	Required or Optional	Available Version
processParameters	<a href="#">Loyalty Process Parameter Map</a>	Input parameters for running the loyalty process.	Required	56.0

## Loyalty Process Input Options

Input representation of the execution options for the loyalty process.

### Properties

Name	Type	Description	Required or Optional	Available Version
executionType	Enum	Specifies the Rules Engine execution type. Possible values are: <ul style="list-style-type: none"> <li>• <b>Explainability</b>—If the execution type is</li> </ul>	Required	56.0



Name	Type	Description	Required or Optional	Available Version
		<p>Explainability, the process is executed and the output parameters are returned in the response.</p> <ul style="list-style-type: none"> <li>Simulation— If the execution type is Simulation, only the simulation details are returned in the response.</li> </ul>		

## Loyalty Process Parameter Map Input

Input representation of the input parameters for the loyalty process.

### Properties

Name	Type	Description	Required or Optional	Available Version
parameterInput	Map<String, Object>	Input parameters for the loyalty process.	Required	56.0

## Loyalty View Transaction Journal History Input

Input representation for getting the transaction journal records of a loyalty member.

### JSON example

```
{
  "membershipNumber": "doc_member1",
  "journalType": "Accrual",
  "journalSubType": "Social",
  "periodStartDate": "2022-01-27T00:00:00Z",
  "periodEndDate": "2022-11-27T00:00:00Z"
}
```

### Properties

Name	Type	Description	Required or Optional	Available Version
journalSubType	String	The subtype of the transaction journal record.	Optional	56.0
journalType	String	<p>The type of the transaction journal.</p> <p>Possible values include:</p> <ul style="list-style-type: none"> <li>Accrual</li> <li>Redemption</li> </ul>	Required	56.0

Name	Type	Description	Required or Optional	Available Version
membershipNumber	String	The membership number of the loyalty program member.	Required	56.0
periodEndDate	String	Retrieve transaction journals until this date.	Optional	56.0
periodStartDate	String	Retrieve transaction journals starting on this date.	Required	56.0

## Redeem Voucher Input


Input representation for voucher redemption.

### JSON example

```
{
  "contactID": "003xx000004WjzAAAS",
  "action": "Reserve",
  "reservationKey": "key1",
  "reservedValue": 50.57,
  "redeemAmount": 50.57,
  "currencyIsoCode": "Yen"
}
```


JSON example to redeem vouchers using `contactId` or `accountID`.

```
{
  "redemptionAmount": "12",
  "contactId": "003xx000004WkhEAAS",
  "accountId": "001xx000003H3NnAAK",
  "memberId": "01Mxx00000002HxEAI"
}
```

 **Note:** Provide `contactId` to redeem the voucher. If `accountID` is provided, then the associated `contactId` is fetched and the voucher is redeemed

### Properties

Name	Type	Description	Required or Optional	Available Version
contactID	String	ID of the contact associated with the account.	Optional	62.0
currencyIsoCode	String	Three-letter ISO 4217 currency code of the monetary currency in which the voucher is redeemed.	Optional	55.0
redeemAmount	Double	The value of the voucher to be redeemed. Specify a value when a member hasn't redeemed the full value of the voucher. This value is only applicable for vouchers	Optional	53.0

Name	Type	Description	Required or Optional	Available Version
		<p>issued using Fixed Value voucher definitions.</p> <p>A value isn't required when the member has redeemed the complete value of the voucher in a single transaction.</p> <p> <b>Note:</b> If you don't specify a value to be redeemed for the fixed-value voucher, the voucher status changes to <i>Redeemed</i>, and the remaining value becomes zero.</p>		
action	ReservationAction	Specify whether to redeem, reserve, or reinstate the voucher. The values for this property could be Reserve, Reinstate, or Redeem.	Required	62.0
reservedValue	Currency	Specify the amount reserved for partially redeemable vouchers.	Optional	62.0
reservationKey	String	Specify the key used to redeem or reinstate a reserved voucher.	Optional	62.0

## Clone Promotion Input

Input representation that decides which existing promotion is cloned and the attributes of the new promotion that's created.

### JSON example

```
{
  "sourcePromotionId": "0c8xx00000000WHAAY",
  "name": "ups clone 4",
  "startDateTime": "2024-01-08T00:00:00.000Z",
  "endDateTime": "2024-01-09T00:00:00.000Z",
  "promotionCode": "sdfgy543wsd",
  "doesReuseEngagementAttributes": "true",
  "additionalFieldValues": {
    "additionalFieldValuesMap": {
      "Description": "test ups clone"
    }
  },
  "relatedRecords": {
    "relatedRecordsList": [
      {
        "relatedObjectApiName": "PromotionChannel",
        "relatedObjectPromotionFieldName": "PromotionId"
      },
      {
        "relatedObjectApiName": "PromotionLoyaltyPtrProd",
        "relatedObjectPromotionFieldName": "PromotionId"
      }
    ]
  }
}
```

```

    },
    {
      "relatedObjectApiName": "PromotionMarketSegment",
      "relatedObjectPromotionFieldName": "PromotionId"
    },
    {
      "relatedObjectApiName": "PromotionProduct",
      "relatedObjectPromotionFieldName": "PromotionId"
    },
    {
      "relatedObjectApiName": "PromotionProductCategory",
      "relatedObjectPromotionFieldName": "PromotionId"
    },
    {
      "relatedObjectApiName": "PromotionLimit",
      "relatedObjectPromotionFieldName": "PromotionId"
    }
  ]
}
}
}

```

### Properties

Name	Type	Description	Required or Optional	Available Version
additionalFieldValues	Map<String, String>	The details of the promotion's additional fields.	Small, 61.0	61.0
cloneEngagementAttributes	Boolean	Indicates whether the new promotion uses the engagement attributes created for the source promotion to track customers' promotion usage.	Small, 61.0	61.0
endDateTime	String	The end date and time of the new promotion.	Small, 61.0	61.0
name	String	The name of the new promotion.	Small, 61.0	61.0
promotionCode	String	The code of the new promotion.	Small, 61.0	61.0
relatedRecords	<a href="#">Related Promotion Records Input []</a> on page 372	The related record for the source promotion that have to be created for the new promotion.	Small, 61.0	61.0
sourcePromotionId	String	The ID of the promotion that's cloned.	Small, 61.0	61.0
startDateTime	String	The start date and time of the new promotion.	Small, 61.0	61.0

## Related Promotion Records Input

Input representation that decides which related records and fields of the existing promotion are cloned and associated with the new promotion.

**Properties**

Name	Type	Description	Required or Optional	Available Version
<del>relatedObjectApiName</del>	String	The API name of the promotion's related object whose records associated with the existing promotion that have to be cloned for the new promotion.	Small, 61.0	61.0
<del>relatedObjectFieldName</del>	String	The name of the Promotion field in the related object whose record has to be cloned and associated with the new promotion.	Small, 61.0	61.0

## Coupon Usage Input

Input representation of Coupon Usage.

**JSON example**

```
{
  "couponCodeList": [
    "QWERTY",
    "ABCD"
  ],
  "activityDateTime": "2024-10-05T12:45:19Z",
  "externalTransactionNumber": "01Vxx000000006T",
  "customerId": "01Mxx0000000001"
}
```

**Properties**

Name	Type	Description	Required or Optional	Available Version
couponCodeList	List<String>	The codes of the list of coupons for which the usage must be increased or decreased.	Required	62.0
activityDateTime	DateTime	The date and time at which the customer redeemed the coupon.	Optional	62.0
<del>externalTransactionNumber</del>	String	The externally generated unique identifier of cart for which the customer redeemed the coupon.	Optional	62.0
customerId	String	The ID of either the contact, the account, or the loyalty program member who redeemed the coupon.	Optional	62.0

## Response Bodies

Here's a list of Loyalty Management API response bodies.

### [Additional Transaction Journal Attributes Output](#)

Output representation of the additional transaction journal fields.

### [Badge Output](#)

Output representation of the badge assigned to a member.

### [Corporate Member Enrollment Output](#)

Output representation of the corporate member enrollment create request.

### [Credit Points Action Output](#)

Output representation of the credit points action type.

### [Debit Points Action Output](#)

Output representation of the debit points action type.

### [Eligible Promotions List](#)

Output representation of a list of eligible promotions.

### [Expression](#)

Output representation of the expression.

### [Expression Value](#)

Output representation of the expression value.

### [Game Assignment Output](#)

Output representation of the details of a game assigned to a member.

### [Game Details](#)

Output representation of the details of a game definition.

### [Game Result](#)

Output representation of the game reward that a participant has won.

### [Game Reward Details](#)

Output representation of the details of the game reward that a participant has won.

### [Game Rewards List](#)

Output representation of the details of a possible game reward.

### [Games List](#)

Output representation of the list of games assigned to a member.

### [Issue Voucher Action Output](#)

Output representation of the issue voucher action type.

### [Individual Member Enrollment Output](#)

Output representation of the individual member enrollment create request.

### [Loyalty Account Output](#)

Represents the details of the loyalty program member's associated account record.

### [Loyalty Engine Realtime Output](#)

Output representation of the loyalty program process result.

[Loyalty Contact Output](#)

Represents the details of the loyalty program member's associated contact record.

[Member Benefits Output](#)

Output representation of all benefits available for a member.

[Member Benefit Output](#)

Output representation of an individual benefit available for a member.

[Member Currency Output](#)

Represents the details of the loyalty program member's associated currencies.

[Member Currency Change](#)

Output representation of the member currency change.

[Member Points Balance](#)

Output representation of the member currency result.

[Member Profile Output](#)

Output representation of the details of a loyalty program member.

[Member Currency](#)

Output representation of the member currency details.

[Member Tier Output](#)

Represents the details of the loyalty program member's associated tiers.

[Member Points Balance Change](#)

Output representation of the points balance change for a member.

[Loyalty Process Output](#)

Output representation of the loyalty process response.

[Loyalty Program Process Rule](#)

Output representation of the loyalty program process rules.

[Loyalty Program Process Rule Step](#)

Output representation of the loyalty program process rule step.

[Loyalty Program Process Rule Action](#)

Output representation of the loyalty program process rule action.

[Loyalty Program Process Rule Action Parameter](#)

Output representation of the loyalty program process rule action parameter.

[Loyalty Program Process Rule Condition](#)

Output representation of the loyalty program process rule condition.

[Loyalty Program Process Rule Condition Filter Criteria](#)

Output representation of the loyalty program process rule condition filter criteria.

[Loyalty Program Process Rule Condition Group](#)

Output representation of the loyalty program process rule condition group.

[Loyalty View Transaction Journal History Output](#)

Output representation of the view transaction history request.

[Participant Game Rewards List](#)

Output representation of the participant game reward object.

[Process Result](#)

Output representation of results of the processed Transaction Journals records.

[Points Change Output](#)

Output representation of the points debited from or credited to the member's points balance by the transaction journal.

[Points Summary](#)

Output representation of the point summary.

[Promotion Details](#)

Output representation of the details of an eligible promotion.

[Promotion Limit](#)

Output representation of the promotion limit of an eligible promotion.

[Promotion Limit Output](#)

Output representation of promotion limits.

[Promotion Reward Details](#)

Output representation of the details of the reward offered by an eligible promotion rule.

[Promotion Rules List](#)

Output representation of an eligible promotion rule.

[Redeem Voucher Output](#)

Represents the response for the redeem voucher POST method.

[Summary](#)

Output representation of the execution summary of the transaction journals.

[Transaction Details Output](#)

Output representation of the transaction journal ledger records.

[Transaction Journal Output](#)

Output representation of the transaction journals.

[Transaction Journal Map Output](#)

Output representation of the loyalty program process simulation result.

[Transaction Journal Result](#)

Output representation of the transaction journals result.

[Transaction Journal Process Result](#)

Output representation of the Transaction Journal result.

[Transaction Ledgers Output](#)

Output representation of the transaction journals and the ledgers associated with the specified loyalty program member.

[Update Cumulative Usage Action Output](#)

Output representation of the update cumulative usage action type.

[Voucher](#)

Output representation of the details of a voucher.

[Vouchers List](#)

Output representation of the list of vouchers associated with the loyalty program member.

[Voucher Result](#)

Output representation of the voucher result.



[Voucher Output](#)

Output representation of the details about a voucher.

[Promotion Configuration Output](#)

Output representation of the promotion's details and its configuration.

[Clone Promotion Output](#)

Output representation of Clone Promotion API.

[Promotion Coupon Availability](#)

Output representation of the promotion coupon availability.

[Promotion Coupon](#)

Output representation of the promotion coupon.

[Coupon Availability Message](#)

Output representation of the coupon availability message.

[Coupon Usage Result](#)

Represents the response for coupon usage increase and decrease POST method.

[Member Engagement Trail Output](#)

Output representation of member engagement trail API.

[Member Engagement Attribute](#)

Output representation of Member Engagement Attribute.

## Additional Transaction Journal Attributes Output

Output representation of the additional transaction journal fields.

Property Name	Type	Description	Filter Group and Version	Available Version
fieldName	String	The name of the transaction journal field.	Small, 58.0	58.0
value	String	The value that's stored in the transaction journal field.	Small, 58.0	58.0

## Badge Output

Output representation of the badge assigned to a member.

Property Name	Type	Description	Filter Group and Version	Available Version
badgeValidityEndDate	String	Date from which the badge is no longer assigned to the member.	Small, 60.0	60.0
loyaltyProgramBadgeName	String	Name of the assigned badge.	Small, 60.0	60.0
memberBadgeStatus	String	Status of the assigned badge.	Small, 60.0	60.0
reason	String	Reason the badge was assigned to the member.	Small, 60.0	60.0

## Corporate Member Enrollment Output

Output representation of the corporate member enrollment create request.

### JSON example

```
* accountId: *001xx000003GgHX*
* loyaltyProgramMemberId: *01Mxx00000008K9*
* loyaltyProgramName: *program1*
* membershipNumber: *Member6*
* transactionJournals
  * [Item 1]
    * activityDate: *2021-01-01T08:00:00.000Z*
    * journalSubType: *Member Enrollment*
    * journalType: *Accrual*
    * loyaltyProgram: *program1*
    * loyaltyProgramMember: *Member6*
    * referredMember: *null*
    * status: *Pending*
    * transactionJournalId: *01Vxx00000002Ej*
  * [Item 2]
    * activityDate: *2021-01-01T08:00:00.000Z*
    * journalSubType: *Member Referral*
    * journalType: *Accrual*
    * loyaltyProgram: *program1*
    * loyaltyProgramMember: *Member1*
    * referredMember: *Member6*
    * status: *Pending*
    * transactionJournalId: *01Vxx00000002Ek*
```

Property Name	Type	Description	Filter Group and Version	Available Version
accountId	String	The ID of the loyalty program member's business account.	Small, 55.0	55.0
loyaltyProgramMemberId	String	The ID of the loyalty program member record created for the member.	Small, 55.0	55.0
loyaltyProgramName	String	The name of the loyalty program into which the member enrolled.	Small, 55.0	55.0
membershipNumber	String	The membership number of the enrolled member.	Small, 55.0	55.0
transactionJournals	<a href="#">Transaction Journal Output[]</a>	The list of transaction journals that are created.	Small, 55.0	55.0

## Credit Points Action Output

Output representation of the credit points action type.

Property Name	Type	Description	Filter Group and Version	Available Version
actionType	String	The type of action in the loyalty program process rule.	Small, 55.0	55.0
changeInEscrow PointsBalance	Double	The change in the escrow points balance of the loyalty program member associated with the transaction journal.	Small, 55.0	55.0
changeInPointsBalance	Double	The change in the points balance of the loyalty program member associated with the transaction journal.	Small, 55.0	55.0
escrowPointsCredit Date	String	The date on which points that are added to the escrow points balance of the loyalty program member associated with the transaction journal must be rolled over to the points balance of the member.	Small, 55.0	55.0
loyaltyProgram CurrencyName	String	The name of the loyalty program currency for which points have been credited to the loyalty program member associated with the transaction journal.	Small, 55.0	55.0
promotionId	String	The ID of the promotion associated with the rule.	Small, 55.0	55.0
stepName	String	The name of the step in the loyalty program process rule.	Small, 55.0	55.0

## Debit Points Action Output

Output representation of the debit points action type.

Property Name	Type	Description	Filter Group and Version	Available Version
actionType	String	The type of action in the loyalty program process rule.	Small, 55.0	55.0
changeInPoints Balance	Double	The change in the points balance of the loyalty program member associated with the transaction journal.	Small, 55.0	55.0
loyaltyProgram CurrencyName	String	The name of the loyalty program currency for which points have been debited from the loyalty program member associated with the transaction journal.	Small, 55.0	55.0
promotionId	String	The ID of the promotion associated with the rule.	Small, 55.0	55.0

Property Name	Type	Description	Filter Group and Version	Available Version
stepName	String	The name of the step in the loyalty program process rule.	Small, 55.0	55.0

## Eligible Promotions List

Output representation of a list of eligible promotions.

### JSON example

```
{
  "promotions": [
    {
      "id": "0c85i0000009c6D",
      "promotionCode": "10OFF",
      "displayName": "10% off on all electronics",
      "priority": 1,
      "currencyCode": "USD",
      "coupon": {
        "couponDetails": {
          "couponCode": "COUPON-123",
          "status": "Active",
          "startDateTime": "2024-01-01",
          "endDateTime": null
        },
        "message": "SingleActive"
      },
      "isAutomatic": false,
      "additionalFields": {
        "startDate": "2023-09-10",
        "endDate": "2023-09-20",
        "promotionLocation__c": "HYD"
      },
      "limits": [
        {
          "type": "PerLineItem",
          "limit": 1
        }
      ],
      "rules": [
        {
          "ruleName": "10PercentElectronics",
          "benefits": [
            {
              "type": "Discount",
              "discountLevel": "Cart",
              "discountType": "Amount",
              "discountValue": 10
            },
            {
              "type": "Discount",
              "discountLevel": "CartLineItem",

```

```

        "lines": [
          {
            "id": "6c139faa-b749-11ed-afa1-0242ac120002",
            "quantity": 3
          },
          {
            "id": "346bded6-b74a-11ed-afa1-0242ac120002",
            "quantity": 2
          }
        ],
        "discountType": "Amount",
        "discountValue": 10
      },
      {
        "type": "Voucher",
        "voucherDefinition": "ELEC10",
        "voucherExpiryDate": "2023-12-31"
      }
    ]
  },
  {
    "ruleName": "5PercentAppliances",
    "benefits": [
      {
        "type": "Discount",
        "discountLevel": "Cart",
        "discountType": "Percent",
        "discountValue": 5
      },
      {
        "type": "FreeProduct",
        "lineItemid": "b749-11ed-afa1",
        "lines": [
          {
            "id": "6c139faa-b749-11ed-afa1-0242ac120002",
            "quantity": 1
          }
        ]
      }
    ]
  }
],
{
  "id": "0c85i0000009c6D",
  "promotionCode": "20OFF",
  "displayName": "Get 100 points on all clothing",
  "priority": 2,
  "currencyCode": "USD",
  "additionalFields": {
    "startDate": "2023-09-10"
  },
  "rules": [
    {

```

```

    "ruleName": "100PointsClothing",
    "benefits": [
      {
        "type": "Points",
        "points": "10",
        "loyaltyProgramCurrencyName": "NQP"
      },
      {
        "type": "Badge",
        "loyaltyProgramBadgeName": "Ranger"
      }
    ]
  }
]
}

```

Property Name	Type	Description	Filter Group and Version	Available Version
Promotions	<a href="#">Promotion Details</a> []	List of eligible promotions for a customer's cart. The list also contains the details of the rewards that each promotion offers to the customer.	Small, 60.0	60.0

## Expression

Output representation of the expression.

Property Name	Type	Description	Filter Group and Version	Available Version
expression	String	The name of an expression.	Small, 55.0	55.0
expressionValue	<a href="#">Expression Value</a>	The values of an expression.	Small, 55.0	55.0
parameterValues	Map<String, <a href="#">Expression Value</a> >	The list of parameter values provided for an expression.	Small, 55.0	55.0
resolvedExpression	String	The resolved fields and values of an expression.	Small, 55.0	55.0

## Expression Value

Output representation of the expression value.

Property Name	Type	Description	Filter Group and Version	Available Version
dataType	ExpressionDataType	The data type of an expression. Possible values are: <ul style="list-style-type: none"> <li>• Boolean</li> <li>• Date</li> <li>• DateTime</li> <li>• Number</li> <li>• Subject</li> <li>• Text</li> </ul>	Small, 55.0	55.0
value	Object	The value of an expression.	Small, 55.0	55.0

## Game Assignment Output

Output representation of the details of a game assigned to a member.

Property Name	Type	Description	Filter Group and Version	Available Version
gameParticipantId	String	ID of the associated game participant.	Small, 60.0	60.0
gameParticipantRewardId	String	The ID of the game participant reward record that stores the details of the reward that the member receives after they play the game. The record is associated with the source transaction journal and the record that issues the reward to the participant.	Small, 60.0	60.0
gameDefinitionName	String	Name of the assigned game definition.	Small, 60.0	60.0
gameExpirationDate	String	Date until which the assigned game is valid.	Small, 60.0	60.0
reason	String	Reason the game was assigned to the member.	Small, 60.0	60.0

## Game Details

Output representation of the details of a game definition.

Property Name	Type	Description	Filter Group and Version	Available Version
description	String	Description of the game.	Small, 60.0	60.0
endDate	String	Date until which participants are eligible to play the game.	Small, 60.0	60.0

Property Name	Type	Description	Filter Group and Version	Available Version
gameRewards	<a href="#">Game Rewards List</a> []	List of possible rewards for a game.	Small, 60.0	60.0
name	String	Name of the game definition.	Small, 60.0	60.0
participantGameRewards	<a href="#">Participant Game Rewards List</a> []	List of participant game reward records that contain the details of the game reward the participant has won for each game play, or can win after playing the game.	Small, 60.0	60.0
startDate	String	Date from which participants are eligible to play the game.	Small, 60.0	60.0
timeoutDuration	Integer	Duration of the game in seconds before the game session times out.	Small, 60.0	60.0
type	String	Type of the game. Possible values are: <ul style="list-style-type: none"> <li>Scratchcard</li> <li>SpintheWheel</li> </ul>	Small, 60.0	60.0

## Game Result

Output representation of the game reward that a participant has won.

### JSON example

Here's an example of a reward won by the participant for the Spin the Wheel game:

```
{
  "errorMessage":null,
  "gameReward":[
    {
      "color":"01CD6C",
      "description":"Loyalty Points",
      "gameRewardId":"3GRSB0000000Caf4AE",
      "imageUrl":null,
      "issuedRewardReference":"01VSB000000032j2AA",
      "name":"1000 Reward Points",
      "rewardType":"LoyaltyPoints",
      "rewardValue":"100.0"
    }
  ],
  "status":true
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
errorMessage	String	Error message that provides the reason why the details of the participant reward weren't found.	Small, 60.0	60.0



Property Name	Type	Description	Filter Group and Version	Available Version
gameReward	<a href="#">Game Reward Details</a> []	Details of the game reward won by the participant.	Small, 60.0	60.0
status	Boolean	Indicates whether the request succeeded ( <code>true</code> ) or failed ( <code>false</code> ).	Small, 60.0	60.0

## Game Reward Details

Output representation of the details of the game reward that a participant has won.

Property Name	Type	Description	Filter Group and Version	Available Version
color	String	Color of the segment in the Spin the Wheel game that represents the game reward.	Small, 60.0	60.0
description	String	Description of the game reward.	Small, 60.0	60.0
gameRewardId	String	ID of the game reward won by the participant.	Small, 60.0	60.0
imageUrl	String	Image URL of the segment in the Spin the Wheel game that represents the game reward.	Small, 60.0	60.0
issuedRewardReference	String	Voucher ID, or the transaction journal ID associated with the voucher, or the loyalty points rewarded to the participant.	Small, 60.0	60.0
name	String	Name of the game reward.	Small, 60.0	60.0
rewardType	String	Type of the game reward. Possible values are: <ul style="list-style-type: none"> <li>• CustomReward</li> <li>• LoyaltyPoints</li> <li>• NoReward</li> <li>• Raffle</li> <li>• Voucher</li> </ul>	Small, 60.0	60.0
rewardValue	String	Value of the voucher, or the number of points to credit as a game reward to participants.	Small, 60.0	60.0

## Game Rewards List

Output representation of the details of a possible game reward.

Property Name	Type	Description	Filter Group and Version	Available Version
color	String	Color of the segment in the Spin the Wheel game that represents the game reward.	Small, 60.0	60.0
description	String	Description of the game reward.	Small, 60.0	60.0
imageUrl	String	Image URL of the segment in the Spin the Wheel game that represents the game reward.	Small, 60.0	60.0
name	String	Name of the game reward.	Small, 60.0	60.0
rewardId	String	ID of the game reward.	Small, 60.0	60.0
rewardType	String	Type of the game reward. Possible values are: <ul style="list-style-type: none"> <li>• CustomReward</li> <li>• LoyaltyPoints</li> <li>• NoReward</li> <li>• Raffle</li> <li>• Voucher</li> </ul>	Small, 60.0	60.0
rewardValue	String	Value of the voucher, or the number of points to credit as a game reward to participants.	Small, 60.0	60.0

## Games List

Output representation of the list of games assigned to a member.

### JSON example

Here's an example of the details of an eligible Spin the Wheel game assigned to a member, and an already played Scratch Card game.

```
{
  "errorMessage":null,
  "gameDefinitions":[
    {
      "description":"Play Spin The Wheel to get amazing rewards",
      "endDate":"2024-07-31T19:00:00.000Z",
      "gameRewards":[
        {
          "color":"FF5733",
          "description":"Get 10% Off Voucher",
          "imageUrl":null,
          "name":"10 % Off",
          "rewardType":"Voucher",
          "rewardValue":null
        },
        {

```

```

        "color":"000000",
        "description":"Loyalty Points",
        "imageUrl":null,
        "name":"1000 Reward Points",
        "rewardType":"LoyaltyPoints",
        "rewardValue":"100.0"
    },
    {
        "color":"762E2E",
        "description":"Better Luck Next Time",
        "imageUrl":null,
        "name":"No Reward",
        "rewardType":"NoReward",
        "rewardValue":null
    }
],
"participantGameRewards":[
    {
        "expirationDate":"2024-01-04T20:00:00.000Z",
        "gameParticipantRewardId":"3PRSB0000000hjJ4AQ",
        "gameRewardId":null,
        "issuedRewardReference":null,
        "sourceActivity":null,
        "status":"YetToReward"
    }
],
"name":"Spin The Wheel",
"startDate":"2023-10-01T19:00:00.000Z",
"timeoutDuration":10,
"type":"SpintheWheel"
},
{
    "description":"Play Scratch card to get amazing rewards",
    "endDate":"2024-07-31T19:00:00.000Z",
    "gameRewards":[
        {
            "color":"",
            "description":"Get 10% Off Voucher",
            "imageUrl":null,
            "name":"10 % Off",
            "rewardType":"Voucher",
            "rewardValue":null
        },
        {
            "color":"",
            "description":"Better Luck Next Time",
            "imageUrl":null,
            "name":"No Reward",
            "rewardType":"NoReward",
            "rewardValue":null
        }
    ],
    "participantGameRewards":[
        {

```

```

        "expirationDate": "2024-01-04T20:00:00.000Z",
        "gameParticipantRewardId": "2ABSB0000000hjJ3AM",
        "gameRewardId": "2GRSB0000000Caf4AB",
        "issuedRewardReference": "01VSB000000032j2AA",
        "sourceActivity": "0c85i0000009c6D",
        "status": "Rewarded"
    }
  ],
  "name": "Scratch Card",
  "startDate": "2023-10-01T19:00:00.000Z",
  "timeoutDuration": 20,
  "type": "Scratchcard"
}
],
"status": true
}

```

Property Name	Type	Description	Filter Group and Version	Available Version
errorMessage	String	Message that provides the reason why a list of games weren't found for the participant.	Small, 60.0	60.0
gameDefinitions	<a href="#">Game Details</a> []	<p>List of game definitions associated with the games assigned to the member.</p> <p>A game definition has the details of a game, including:</p> <ul style="list-style-type: none"> <li>The list of possible rewards for the game.</li> <li>The participant game reward records that contain details of the game reward the participant has won for each game play, or can win after playing the game.</li> </ul>	Small, 60.0	60.0
status	Boolean	Indicates whether the request succeeded ( <code>true</code> ) or failed ( <code>false</code> ).	Small, 60.0	60.0

## Issue Voucher Action Output

Output representation of the issue voucher action type.

Property Name	Type	Description	Filter Group and Version	Available Version
actionType	String	The type of action in the loyalty program process rule.	Small, 55.0	55.0
discountPercent	Integer	The discount percentage provided by the voucher.	Small, 55.0	55.0
effectiveDate	String	The date the voucher goes into effect.	Small, 55.0	55.0

Property Name	Type	Description	Filter Group and Version	Available Version
expiryDate	String	The date the voucher expires.	Small, 55.0	55.0
faceValue	Double	The value of the voucher.	Small, 55.0	55.0
productId	String	The ID of the product associated with the voucher.	Small, 55.0	55.0
promotionId	String	The ID of the promotion associated with the rule.	Small, 55.0	55.0
stepName	String	The name of the step in the loyalty program process rule.	Small, 55.0	55.0
voucherCode	String	The code of the voucher.	Small, 55.0	55.0
voucherDefinition Name	String	The name of the voucher definition associated with the voucher code.	Small, 55.0	55.0
voucherId	String	The ID of the voucher.	Small, 55.0	55.0

## Individual Member Enrollment Output

Output representation of the individual member enrollment create request.

### JSON example

```
* contactId: *003xx000004Ws08*
* loyaltyProgramMemberId: *01Mxx00000008FJ*
* loyaltyProgramName: *program1*
* membershipNumber: *Member3*
* transactionJournals
  * [Item 1]
    * activityDate: *2021-01-01T08:00:00.000Z*
    * journalSubType: *Member Enrollment*
    * journalType: *Accrual*
    * loyaltyProgram: *program1*
    * loyaltyProgramMember: *Member3*
    * referredMember: *null*
    * status: *Pending*
    * transactionJournalId: *01Vxx000000029t*
  * [Item 2]
    * activityDate: *2021-01-01T08:00:00.000Z*
    * journalSubType: *Member Referral*
    * journalType: *Accrual*
    * loyaltyProgram: *program1*
    * loyaltyProgramMember: *Member1*
    * referredMember: *Member3*
    * status: *Pending*
    * transactionJournalId: *01Vxx000000029u*
```

Property Name	Type	Description	Filter Group and Version	Available Version
contactId	String	The ID of the loyalty program member's person account.	Small, 55.0	55.0
loyaltyProgram MemberId	String	The ID of the loyalty program member record created for the member.	Small, 55.0	55.0
loyaltyProgram Name	String	The name of the loyalty program into which the member is enrolled.	Small, 55.0	55.0
membershipNumber	String	The membership number of the enrolled member.	Small, 55.0	55.0
transactionJournals	<a href="#">Transaction Journal Output[]</a>	The list of transaction journals that are created.	Small, 55.0	55.0

## Loyalty Account Output

Represents the details of the loyalty program member's associated account record.

Property Name	Type	Description	Filter Group and Version	Available Version
accountId	String	The ID of the associated account record.	Small, 55.0	55.0
name	String	The name of the associated account.	Small, 55.0	55.0

## Loyalty Engine Realtime Output

Output representation of the loyalty program process result.

### JSON example


```
{
  "message":null,
  "processResult":{
    "memberPointsBalance":[
      {
        "memberCurrencies":[
          {
            "currencyName":"Tier Coins",
            "currencyType":"Qualifying",
            "escrowPointsBalance":0.0,
            "pointsBalance":350.0
          },
          {
            "currencyName":"Coins",
            "currencyType":"NonQualifying",
            "escrowPointsBalance":0.0,
            "pointsBalance":300.0
          }
        ]
      }
    ]
  }
}
```

```

        ],
        "memberShipNumber":"M5"
    }
],
"transactionJournalResult":[
    {
        "badgesAssigned":[
            {
                "badgeValidityEndDate":"2023-12-15",
                "loyaltyProgramBadgeName":"Gen AI Badge",
                "memberBadgeStatus":"1",
                "reason":"Test"
            }
        ],
        "changeInMemberPoints":[
            {
                "changeInMemberCurrencies":[
                    {
                        "changeInPointsBalance":100.0,
                        "currencyName":"Coins",
                        "currencyType":"NonQualifying",
                        "promotionId":"0c8RM0000004Fix"
                    }
                ],
                "memberShipNumber":"M5"
            }
        ],
        "gamesAssigned":[
            {
                "gameDefinitionName":"3PRRM0000004C98",
                "gameExpirationDate":"2023-12-28T00:00:00.000Z",
                "reason":"Test"
            }
        ],
        "id":"01VRM0000002qnB2AQ",
        "voucherResult":{
            "vouchersIssued":[
                {
                    "discountPercent":10,
                    "effectiveDate":"2023-11-29",
                    "expirationDate":"2023-12-15",
                    "promotionId":"0c8RM0000004Fix",
                    "voucherDefinitionName":"10% Discount Voucher"
                }
            ]
        }
    }
],
"programName":"NTO",
"simulationResults":null,
"status":true
}

```

Property Name	Type	Description	Filter Group and Version	Available Version
message	String	Error message.	Small, 54.0	54.0
processResult	<a href="#">Process Result</a>	Details of each transaction journal in a request, and also the member points balance that is the currency-wise balances after all the transaction journals are processed.  The process result collection is empty when the simulation is run.	Small, 54.0	54.0
programName	String	Name of the loyalty program.	Small, 55.0	55.0
simulationResults	<a href="#">Transaction Journal Map Output</a>	Simulation result that details how the loyalty program process rules can process a transaction journal.  The simulation result collection is empty when the loyalty program process is run.	Small, 55.0	55.0
status	Boolean	Indicates whether the request succeeded ( <code>true</code> ) or failed ( <code>false</code> ).	Small, 54.0	54.0

 **Important:** The response contains only the details of the points, badges, games, and vouchers that are provided by the loyalty program process rules executed for the member's order. The response doesn't contain the details of the rewards provided by the flows or the subprocesses that the loyalty program process rules execute.

## Loyalty Contact Output

Represents the details of the loyalty program member's associated contact record.

Property Name	Type	Description	Filter Group and Version	Available Version
contactId	String	The ID of the associated contact record.	Small, 55.0	55.0
email	String	The email of the associated contact.	Small, 55.0	55.0
firstName	String	The first name of the associated contact.	Small, 55.0	55.0
lastName	String	The last name of the associated contact.	Small, 55.0	55.0

## Member Benefits Output

Output representation of all benefits available for a member.



## JSON Example

```
{
  "result" : {
    "benefitId": "01B000000abc123",
    "benefitName": "15 Kg Luggage",
    "benefitTypeId": "01o000000afja435",
    "benefitTypeName": "Carry On Luggage",
    "startDate": "23-05-2015",
    "endDate": "22-05-2035",
    "isActive": true,
    "createdRecordId": "570000000ak123",
    "createdRecordName": "15 Kg Luggage Carry On",
    "description": "This is a carry on luggage member benefit"
  },
  {
    "benefitId": "01B000000gsy88",
    "benefitName": "Priority Support",
    "benefitTypeId": "01o000000pi23ii",
    "benefitTypeName": "Support Entitlements",
    "startDate": "16-08-2020",
    "endDate": "15-02-2021",
    "isActive": true,
    "createdRecordId": "550000000aj112",
    "createdRecordName": "Priority Support Entitlement",
    "description": "This is a support entitlement member benefit"
  }
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
memberBenefits	<a href="#">Member Benefit Output[]</a>	List of all benefits available for a member.	Small, 51.0	51.0

## Member Benefit Output

Output representation of an individual benefit available for a member.

Property Name	Type	Description	Filter Group and Version	Available Version
benefitId	String	The ID of the benefit available for the member.	Small, 51.0	51.0
benefitName	String	The name of the benefit available for the member.	Small, 51.0	51.0
benefitTypeId	String	The ID of the associated benefit type record.	Small, 51.0	51.0
benefitTypeName	String	The name of the associated benefit type.	Small, 51.0	51.0

Property Name	Type	Description	Filter Group and Version	Available Version
createdRecordId	String	The ID of the record created during benefit assignment.	Small, 51.0	51.0
createdRecordName	String	The display name of the record created during benefit assignment.	Small, 51.0	51.0
description	String	The description of the benefit available to a loyalty program member.	Small, 58.0	58.0
endDate	String	The date the benefit expires.	Small, 51.0	51.0
isActive	Boolean	Indicates whether the benefit is active ( <code>true</code> ) or not ( <code>false</code> ).	Small, 51.0	51.0
startDate	String	The date from when the benefit applies to a member.	Small, 51.0	51.0

## Member Currency Output

Represents the details of the loyalty program member's associated currencies.

Property Name	Type	Description	Filter Group and Version	Available Version
additionalLoyaltyMemberCurrencyFields	Map<String, Object>	The details of the custom fields added in the loyalty member currency object.	Small, 55.0	55.0
escrowPointsBalance	Double	The escrow points balance of the member for the currency.	Small, 55.0	55.0
expirablePoints	Double	The number of points that the member hasn't redeemed for the currency.	Small, 55.0	55.0
lastAccrualProcessedDate	String	The date on which the member's accruals for the currency were last processed.	Small, 55.0	55.0
lastEscrowProcessedDate	String	The date on which the member's escrow points were last rolled over to their points balance for the currency.	Small, 55.0	55.0
lastExpirationProcessRunDate	String	The date on which the member's points were last expired for the currency.	Small, 55.0	55.0
lastPointsAggregationDate	String	The date on which the member's points for the fixed type non-qualifying currency were last aggregated.	Small, 55.0	55.0
lastPointsResetDate	String	The date on which the member's points for the currency were last reset.	Small, 55.0	55.0
loyaltyMemberCurrencyName	String	The name of the Loyalty Member Currency record.	Small, 55.0	55.0

Property Name	Type	Description	Filter Group and Version	Available Version
loyaltyProgramCurrencyId	String	The ID of the loyalty program currency associated with the member.	Small, 55.0	55.0
loyaltyProgramCurrencyName	String	The name of the loyalty program currency associated with the member.	Small, 55.0	55.0
memberCurrencyId	String	The ID of the Loyalty Member Currency record.	Small, 55.0	55.0
nextQualifyingPointsResetDate	String	The date on which the member's points balance for the qualifying currency is reset next.	Small, 55.0	55.0
pointsBalance	Double	The points balance of the member for the currency.	Small, 55.0	55.0
qualifyingPointsBalanceBeforeReset	Double	The last point balance of the member before the member's points balance was reset to zero for the qualifying currency.	Small, 55.0	55.0
totalEscrowPointsAccrued	Double	The total number of points that the member has accrued in escrow for the currency.	Small, 55.0	55.0
totalEscrowRolloverPoints	Double	The total number of points that has rolled over from the member's escrow point balance to the member's points balance for the currency.	Small, 55.0	55.0
totalPointsAccrued	Double	The total number of points the member has accrued for the currency.	Small, 55.0	55.0
totalPointsExpired	Double	The total number of points that have expired for the member.	Small, 55.0	55.0
totalPointsRedeemed	Double	The total number of points the member has redeemed for the currency.	Small, 55.0	55.0

## Member Currency Change

Output representation of the member currency change.

Property Name	Type	Description	Filter Group and Version	Available Version
changeEscrowPointsBalance	Double	The change in escrow points balance.	Small, 54.0	54.0
changeInPointsBalance	Double	The change in points balance.	Small, 54.0	54.0
currencyName	String	The name of the member currency.	Small, 54.0	54.0

Property Name	Type	Description	Filter Group and Version	Available Version
currencyType	String	The type of member currency. Possible values are: <ul style="list-style-type: none"> <li>NonQualifying</li> <li>Qualifying</li> </ul>	Small, 54.0	54.0
promotionId	String	The promotion ID for which the member is credited points.	Small, 54.0	54.0

## Member Points Balance

Output representation of the member currency result.

Property Name	Type	Description	Filter Group and Version	Available Version
memberCurrencies	<a href="#">Member Currency</a> []	List of currencies associated with the loyalty program member.	Small, 54.0	54.0
memberShipNumber	String	The membership number of the loyalty program member who is associated with the transaction journal that's created.	Small, 54.0	54.0

## Member Profile Output

Output representation of the details of a loyalty program member.

### JSON example

```
{
  "programName": "baLoyalty",
  "associatedContactDetails": {
    "id": "003xx0000000003",
    "firstName": "Abcd",
    "lastName": "Xyz",
    "email": "abcd@salesforce.com"
  },
  "associatedAccountDetails": {
    "id": "001xx0000000003",
    "name": "Salesforce"
  },
  "membershipNumber": "1234",
  "memberId": "01Mxx0000A000001EAA",
  "memberStatus": "Active",
  "memberType": "Individual",
  "groupName": "Family Group",
  "groupCreatedByMember": "GM-001",
  "associatedCorporateMembershipNumber": "G-001",
  "referredBy": "H-001",
}
```

```

"lastActivityDate": "2022-01-02",
"statementLastGeneratedDate": "2022-09-09",
"statementFrequency": "Monthly",
"statementMethod": "Email",
"enrollmentDate": "2022-01-01",
"enrollmentChannel": "POS",
"canReceivePromotions": "True",
"canReceivePartnerPromotion": "True",
"membershipEndDate": "2023-01-01",
"membershipLastRenewalDate": "2021-01-01",
"memberCurrencies": [
  {
    "programCurrency": "NQP",
    "pointsBalance": 500,
    "escrowPointsBalance": 25
  }
],
"memberTiers": [
  {
    "tierGroupName": "QualifyingGroup",
    "tierName": "Gold",
    "sequenceNumber": "4",
    "effectiveDate": "2022-01-01",
    "expirationDate": "2022-01-10"
  }
],
"additionalMemberFieldValues": {
  "secondaryEmail__c": "abcd@salesforce.com"
},
"additionalMemberCurrencyFieldValues": {
  "conversionRate__c": "5"
},
"additionalMemberTierFieldValues": {
  "previousTier__c": "Silver"
}
}

```

Property Name	Type	Description	Filter Group and Version	Available Version
additionalLoyaltyProgramMemberFields	Map<String, Object>	The details of the custom fields added in the loyalty program member object.	Small, 55.0	55.0
associatedAccount	<a href="#">Member Account Output</a>	The details of the account record that is associated with the loyalty program member.	Small, 55.0	55.0
associatedContact	<a href="#">Member Contact Output</a>	The details of the contact record that is associated with the loyalty program member.	Small, 55.0	55.0
canReceivePartnerPromotions	Boolean	Indicates whether the loyalty program member can receive partner promotions.	Small, 55.0	55.0

Property Name	Type	Description	Filter Group and Version	Available Version
canReceivePromotions	Boolean	Indicates whether the loyalty program member can receive promotions ( <code>true</code> ) or not ( <code>false</code> ).	Small, 55.0	55.0
enrollmentChannel	String	The channel used by the member to enroll in the loyalty program.	Small, 55.0	55.0
enrollmentDate	String	The date on which the member enrolled in the loyalty program.	Small, 55.0	55.0
groupCreatedByMember	String	The membership number of the loyalty program member who created the group.	Small, 55.0	55.0
groupName	String	The name of the loyalty program group to which the member belongs.	Small, 55.0	55.0
lastActivityDate	String	The date on which the member performed their last activity.	Small, 55.0	55.0
loyaltyProgramMemberId	String	The ID of the loyalty program member.	Small, 55.0	55.0
loyaltyProgramName	String	The name of the loyalty program.	Small, 55.0	55.0
memberCurrencies	<a href="#">Member Currency Output[]</a>	The currencies associated with the loyalty program member.	Small, 55.0	55.0
memberStatus	String	The status of the loyalty program member.	Small, 55.0	55.0
memberTiers	<a href="#">Member Tier Output[]</a>	The tiers associated with the loyalty program member.	Small, 55.0	55.0
memberType	String	The type of loyalty program member.	Small, 55.0	55.0
membershipEndDate	String	The date on which the membership ends.	Small, 55.0	55.0
membershipLastRenewalDate	String	The date on which the loyalty program member last renewed their membership.	Small, 55.0	55.0
membershipNumber	String	The membership number of the loyalty program member.	Small, 55.0	55.0
referredBy	String	The membership number of the loyalty program member who referred the member to the loyalty program.	Small, 55.0	55.0
relatedCorporateMembershipNumber	String	The membership number of the Corporate type loyalty program member associate with the member.	Small, 55.0	55.0

Property Name	Type	Description	Filter Group and Version	Available Version
transactionJournalStatementFrequency	String	The frequency at which transaction journal statements must be delivered to the member.  Possible values are: <ul style="list-style-type: none"> <li>• Monthly</li> <li>• Quarterly</li> </ul>	Small, 55.0	55.0
transactionJournalStatementCreateDate	String	The date on which the member's last transaction journal statement was generated.	Small, 55.0	55.0
transactionJournalStatementMethod	String	The method used to deliver transaction journal statements to the member.	Small, 55.0	55.0

## Member Currency

Output representation of the member currency details.

Property Name	Type	Description	Filter Group and Version	Available Version
currencyName	String	The name of the member currency.	Small, 54.0	54.0
currencyType	String	The type of member currency.  Possible values are: <ul style="list-style-type: none"> <li>• NonQualifying</li> <li>• Qualifying</li> </ul>	Small, 54.0	54.0
escrowPointsBalance	Double	The number of points the member has in escrow for a currency.	Small, 54.0	54.0
pointsBalance	Double	The member's points balance for a currency.	Small, 54.0	54.0

## Member Tier Output

Represents the details of the loyalty program member's associated tiers.

Property Name	Type	Description	Filter Group and Version	Available Version
additionalTierFields	Map<String, Object>	The details of the custom fields added in the Loyalty Member Tier object.	Small, 55.0	55.0
areTierBenefitsAssigned	Boolean	Indicates whether the benefits associated with the tier have been assigned to the member ( <code>true</code> ) or not ( <code>false</code> ).	Small, 55.0	55.0

Property Name	Type	Description	Filter Group and Version	Available Version
loyaltyMemberTierId	String	The ID of tier associated with the loyalty program member.	Small, 55.0	55.0
loyaltyMemberTierName	String	The name of the loyalty member tier.	Small, 55.0	55.0
tierChangeReason	String	The reason why the member's was assigned the current tier.	Small, 55.0	55.0
tierChangeReasonType	String	The reason type why the member's was assigned the current tier.	Small, 55.0	55.0
tierEffectiveDate	String	The date on which the member was assigned the tier.	Small, 55.0	55.0
tierExpirationDate	String	The date on which the member's current tier expires.	Small, 55.0	55.0
tierGroupId	String	The ID of the tier group associated with the Loyalty Tier record.	Small, 55.0	55.0
tierGroupName	String	The name of the tier group.	Small, 55.0	55.0
tierId	String	The ID of the Loyalty Tier record.	Small, 55.0	55.0
tierSequenceNumber	Integer	The sequence number of the tier.	Small, 55.0	55.0

## Member Points Balance Change


Output representation of the points balance change for a member.

Property Name	Type	Description	Filter Group and Version	Available Version
changeInMemberCurrencies	<a href="#">Member Currency Change[]</a>	List of points balance changes for the member's currencies based on the transaction journal and promotions.	Small, 54.0	54.0
membershipNumber	String	The membership number of the loyalty program member who is associated with the transaction journal that's created.	Small, 54.0	54.0

## Loyalty Process Output

Output representation of the loyalty process response.



Property Name	Type	Description	Filter Group and Version	Available Version
message	String	The message that provides details of the error in executing the loyalty process request.	Small, 56.0	56.0
outputParameters	Map<String, Object>	The parameters returned after the execution of a loyalty process.	Small, 56.0	56.0
simulationDetails	Map<String, Object>	Specifies the simulation execution type details including detailed steps on how the process is executed.	Small, 56.0	56.0
		 <b>Note:</b> This property is for future use.		
status	Boolean	Indicates whether the request succeeded ( <code>true</code> ) or failed ( <code>false</code> ).	Small, 56.0	56.0

## Loyalty Program Process Rule

Output representation of the loyalty program process rules.

Property Name	Type	Description	Filter Group and Version	Available Version
actionResultSummary	Various	<p>The result provided by each action that processed the transaction journals. The type of action must be configured in the loyalty program process rule.</p> <ul style="list-style-type: none"> <li>• <code>CreditPoints</code>—Credit loyalty points to a specified loyalty program member's balance. See <a href="#">Credit Points Action Output</a>.</li> <li>• <code>DebitPoints</code>—Debit loyalty points from a specified loyalty program member's balance. See <a href="#">Debit Points Action Output</a>.</li> <li>• <code>IssueVoucher</code>—Issue a voucher for a loyalty program member. See <a href="#">Issue Voucher Action Output</a>.</li> <li>• <code>UpdateUsageForCumulativePromotion</code>—Update the progress made by a loyalty program member towards attaining a cumulative type promotion. See <a href="#">Update Cumulative Action Output</a>.</li> </ul>	Small, 55.0	55.0
errorMessage	String	The message stating the reason for error, if any.	Small, 55.0	55.0

Property Name	Type	Description	Filter Group and Version	Available Version
executionStatus	String	The execution status of the loyalty program process rule. Valid values are: <ul style="list-style-type: none"> <li>CriteriaNotMet</li> <li>Failed</li> <li>Success</li> </ul>	Small, 55.0	55.0
executionSummary	Summary	The summary of how the loyalty program process rules processed the transaction journals.	Small, 55.0	55.0
previousRuleName	String	The name of the loyalty program process rule that processed the transaction journals before the current loyalty program process rule.	Small, 55.0	55.0
processRuleName	String	The name of the loyalty program process rule that processed the transaction journals.	Small, 55.0	55.0
promotionName	String	The name of the loyalty program promotion associated with the rule.	Small, 55.0	55.0
ruleSteps	Various	A list of steps that are part of the loyalty process process rules. <ul style="list-style-type: none"> <li>Action—Loyalty program process rule action. See <a href="#">Loyalty Program Process Rule Action</a>.</li> <li>Condition—Loyalty program process rule condition. See <a href="#">Loyalty Program Process Rule Condition</a>.</li> <li>ConditionGroup—Loyalty program process rule condition group. See <a href="#">Loyalty Program Process Rule Condition Group</a>.</li> <li>DefaultPath</li> </ul>	Small, 55.0	55.0

## Loyalty Program Process Rule Step

Output representation of the loyalty program process rule step.

Property Name	Type	Description	Filter Group and Version	Available Version
errorMessage	String	The message stating the reason for error, if any.	Small, 55.0	55.0

Property Name	Type	Description	Filter Group and Version	Available Version
sequenceNumber	Integer	The sequence number of a rule step.	Small, 55.0	55.0
status	String	The status of the loyalty program process rule step execution. Possible values are: <ul style="list-style-type: none"> <li>CriteriaNotMet</li> <li>Failed</li> <li>Success</li> </ul>	Small, 55.0	55.0
stepName	String	The name of the step in the loyalty program process rule.	Small, 55.0	55.0
stepType	String	Specifies the type of step in the loyalty program process rule. Possible values are: <ul style="list-style-type: none"> <li>Action</li> <li>Condition</li> <li>ConditionGroup</li> <li>DefaultPath</li> </ul>	Small, 55.0	55.0

## Loyalty Program Process Rule Action

Output representation of the loyalty program process rule action.

Property Name	Type	Description	Filter Group and Version	Available Version
actionParameters	<a href="#">Loyalty Program Process Rule Action Parameter[]</a>	The list of parameters of the action.	Small, 55.0	55.0
actionType	String	The type of action in the loyalty program process rule. Possible values are: <ul style="list-style-type: none"> <li>AssignParameterValues</li> <li>CreditPoints</li> <li>DeditPoints</li> <li>GetMemberAttributesValues</li> <li>GetMemberPointBalance</li> <li>GetMemberTier</li> <li>GetOutputsFromDecisionTable</li> <li>IncreaseUsageForCumulativePromotion</li> </ul>	Small, 55.0	55.0

Property Name	Type	Description	Filter Group and Version	Available Version
		<ul style="list-style-type: none"> <li>IssueVoucher</li> <li>RunFlow</li> <li>UpdateCurrentValueForMemberAttribute</li> <li>UpdateUsageForCumulativePromotion</li> </ul>		
decisionTableName	String	The name of the decision table being executed with the action.	Small, 55.0	55.0
errorMessage	String	The message stating the reason for error, if any.	Small, 55.0	55.0
flowDefinitionName	String	The name of the flow definition that is run with the action.	Small, 55.0	55.0
sequenceNumber	Integer	The sequence number of a rule step in the process rule.	Small, 55.0	55.0
status	String	<p>The status of the loyalty program process rule step execution.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>CriteriaNotMet</li> <li>Failed</li> <li>Success</li> </ul>	Small, 55.0	55.0
stepName	String	The name of the step in the loyalty program process rule.	Small, 55.0	55.0
stepType	String	<p>Specifies the type of step in the loyalty program process rule.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>Action</li> <li>Condition</li> <li>ConditionGroup</li> <li>DefaultPath</li> </ul>	Small, 55.0	55.0

## Loyalty Program Process Rule Action Parameter

Output representation of the loyalty program process rule action parameter.

Property Name	Type	Description	Filter Group and Version	Available Version
name	String	The name of the parameter.	Small, 55.0	55.0

Property Name	Type	Description	Filter Group and Version	Available Version
operator	String	The type of operator used in the action. Possible value is: <ul style="list-style-type: none"> <li>Equals</li> </ul>	Small, 55.0	55.0
sequenceNumber	Integer	The sequence number of the parameter in the action.	Small, 55.0	55.0
value	Expression	The value of the parameter.	Small, 55.0	55.0

## Loyalty Program Process Rule Condition

Output representation of the loyalty program process rule condition.

Property Name	Type	Description	Filter Group and Version	Available Version
childSteps	<a href="#">Loyalty Program Process Rule Step</a> []	The child steps of the condition.	Small, 55.0	55.0
errorMessage	String	The message stating the reason for error, if any.	Small, 55.0	55.0
filterCondition	String	The logic that is specified to apply the filter conditions.	Small, 55.0	55.0
filterCriteria	<a href="#">Loyalty Program Process Rule Condition Filter Criteria</a> []	The filter criteria of a condition.	Small, 55.0	55.0
isConditionMet	Boolean	Indicates whether the transaction journal meets this condition ( <code>true</code> ) or not ( <code>false</code> ).	Small, 55.0	55.0
sequenceNumber	Integer	Sequence number of a rule step in a rule.	Small, 55.0	55.0
status	String	The status of the loyalty program process rule step execution. Possible values are: <ul style="list-style-type: none"> <li>CriteriaNotMet</li> <li>Failed</li> <li>Success</li> </ul>	Small, 55.0	55.0
stepName	String	The name of the step in the loyalty program process rule.	Small, 55.0	55.0

Property Name	Type	Description	Filter Group and Version	Available Version
stepType	String	Specifies the type of step in the loyalty program process rule.  Possible values are: <ul style="list-style-type: none"> <li>Action</li> <li>Condition</li> <li>ConditionGroup</li> <li>DefaultPath</li> </ul>	Small, 55.0	55.0

## Loyalty Program Process Rule Condition Filter Criteria

Output representation of the loyalty program process rule condition filter criteria.

Property Name	Type	Description	Filter Group and Version	Available Version
filterCriteria Result	Boolean	Indicates whether the transaction journal meets this filter criteria ( <code>true</code> ) or not ( <code>false</code> ).	Small, 55.0	55.0
operator	String	The operator of a condition filter criteria.  Possible values are: <ul style="list-style-type: none"> <li>Contains</li> <li>DoesNotContain</li> <li>Equals</li> <li>GreaterThan</li> <li>GreaterThanOrEquals</li> <li>IsNotNull</li> <li>IsNull</li> <li>LessThan</li> <li>LessThanOrEquals</li> <li>NotEquals</li> </ul>	Small, 55.0	55.0
sequence	Integer	The sequence number of the criteria in a rule condition.	Small, 55.0	55.0
sourceFieldName	Expression	The source field of a condition or action.	Small, 55.0	55.0
value	Expression	The value of an expression.	Small, 55.0	55.0

Property Name	Type	Description	Filter Group and Version	Available Version
valueType	String	The type of value that is specified for a condition or an action parameter.  Possible values are: <ul style="list-style-type: none"> <li>FixedValue</li> <li>Formula</li> <li>Lookup</li> <li>Parameter</li> </ul>	Small, 55.0	55.0

## Loyalty Program Process Rule Condition Group

Output representation of the loyalty program process rule condition group.

Property Name	Type	Description	Filter Group and Version	Available Version
childSteps	<a href="#">Loyalty Program Process Rule Step</a> []	The child steps of the condition group.	Small, 55.0	55.0
errorMessage	String	The message stating the reason for error, if any.	Small, 55.0	55.0
sequenceNumber	Integer	The sequence number of a rule step in a rule.	Small, 55.0	55.0
status	String	The status of the loyalty program process rule step execution.  Possible values are: <ul style="list-style-type: none"> <li>CriteriaNotMet</li> <li>Failed</li> <li>Success</li> </ul>	Small, 55.0	55.0
stepName	String	The name of the step in the loyalty program process rule.	Small, 55.0	55.0
stepType	String	Specifies the type of step in the loyalty program process rule.  Possible values are: <ul style="list-style-type: none"> <li>Action</li> <li>Condition</li> <li>ConditionGroup</li> <li>DefaultPath</li> </ul>	Small, 55.0	55.0

## Loyalty View Transaction Journal History Output

Output representation of the view transaction history request.

### JSON example

```
{
  "message":null,
  "status":true,
  "totalCount":2,
  "transactionJournals":[
    {
      "ActivityDate":"2021-11-27T00:00:00Z",
      "BookedFareClass":"",
      "BookedRoomType":"",
      "BookingDate":"2020-10-23T00:00:00Z",
      "Brand":"Ikea",
      "Channel":"Offline",
      "Destination":"London",
      "Distance":0,
      "EndDate":"2021-10-23T00:00:00Z",
      "ErrorDescription":"",
      "EscrowPointsCreditDate":"2020-10-23",
      "Establishment":"Build",
      "ExternalTransactionNumber":"1",
      "FlightNumber":"123456",
      "Industry":"Wellness",
      "InvoiceDate":"2020-10-23",
      "JournalDate":"2021-12-25T00:00:00Z",
      "JournalReason":"",
      "JournalSubTypeId":"01ST10000000045MAA",
      "JournalTypeId":"01ET1000000009GMAQ",
      "LoyaltyProgramId":"01pT100000000HbIAI",
      "MemberId":"01MT100000000ZLMAY",
      "OrderId":"",
      "OrderItemId":"",
      "Origin":"NewYork",
      "PartnerId":"",
      "PaymentMethod":"",
      "ProductCategoryId":"0ZGT100000000AsOAI",
      "ProductId":"01tT1000000EQcXIAW",
      "PromotionId":"",
      "Quantity":1,
      "QuantityUnitOfMeasureId":"",
      "ReferredMemberId":"",
      "RelatedCorporateMembershipId":"",
      "RelatedJournalId":"",
      "StartDate":"2020-10-23T00:00:00Z",
      "TransactionAmount":45,
      "TransactionLocation":"",
      "TraveledFareClass":"",
      "VoucherCode":""
    },
    {
      "ActivityDate":"2021-11-27T00:00:00Z",
```



```

    "BookedFareClass": "",
    "BookedRoomType": "",
    "BookingDate": "2020-10-23T00:00:00Z",
    "Brand": "Ikea",
    "Channel": "Offline",
    "Destination": "London",
    "Distance": 0,
    "EndDate": "2021-10-23T00:00:00Z",
    "ErrorDescription": "",
    "EscrowPointsCreditDate": "2020-10-23",
    "Establishment": "Build",
    "ExternalTransactionNumber": "1",
    "FlightNumber": "123456",
    "Industry": "Wellness",
    "InvoiceDate": "2020-10-23",
    "JournalDate": "2021-12-25T00:00:00Z",
    "JournalReason": "",
    "JournalSubTypeId": "01ST10000000045MAA",
    "JournalTypeId": "01ET1000000009GMAQ",
    "LoyaltyProgramId": "01pT100000000HbIAI",
    "MemberId": "01MT100000000ZLMAY",
    "OrderId": "",
    "OrderItemId": "",
    "Origin": "NewYork",
    "PartnerId": "",
    "PaymentMethod": "",
    "ProductCategoryId": "0ZGT100000000AsOAI",
    "ProductId": "01tT1000000EQcXIAW",
    "PromotionId": "",
    "Quantity": 2,
    "QuantityUnitOfMeasureId": "",
    "ReferredMemberId": "",
    "RelatedCorporateMembershipId": "",
    "RelatedJournalId": "",
    "StartDate": "2020-10-23T00:00:00Z",
    "TransactionAmount": 55,
    "TransactionLocation": "",
    "TraveledFareClass": "",
    "VoucherCode": ""
  }
]
}

```

Property Name	Type	Description	Filter Group and Version	Available Version
message	String	The message explaining why the request failed.	Small, 56.0	56.0
status	Boolean	Indicates whether the request succeeded ( <code>true</code> ) or failed ( <code>false</code> ).	Small, 56.0	56.0
totalCount	Integer	The total number of transaction journals that match the input criteria.	Small, 56.0	56.0

Property Name	Type	Description	Filter Group and Version	Available Version
transactionJournals	Map<String, Object>[]	The details of the transaction journals.	Small, 56.0	56.0

## Participant Game Rewards List

Output representation of the participant game reward object.

Property Name	Type	Description	Filter Group and Version	Available Version
expirationDate	String	Date until which the game reward is valid.	Small, 60.0	60.0
gameParticipantRewardId	String	ID of the participant game reward object.	Small, 60.0	60.0
gameRewardId	String	ID of the game reward won. This field is null for a game yet to be played.	Small, 60.0	60.0
issuedRewardReference	String	Voucher ID, or the transaction journal ID associated with the voucher, or the loyalty points rewarded to the participant. This field is null for a game yet to be played.	Small, 60.0	60.0
sourceActivity	String	ID of the promotion, or the transaction journal that is associated with the activity because of which the participant was eligible to play the game. Examples of source activities include purchasing a product, and reviewing on social media. This field is null for a game yet to be played.	Small, 60.0	60.0
status	String	Status of the reward. Possible values are: <ul style="list-style-type: none"> <li>Expired</li> <li>NoReward</li> <li>Rewarded</li> <li>YetToReward</li> </ul>	Small, 60.0	60.0

## Process Result

Output representation of results of the processed Transaction Journals records.

Property Name	Type	Description	Filter Group and Version	Available Version
memberPointsBalance	Member Points Balance[]	List of member currencies for which points balance has changed based on the Transaction Journal records that are created.	Small, 54.0	54.0

Property Name	Type	Description	Filter Group and Version	Available Version
<del>transactionJournalResult</del>	Transaction Journal Result[]	Details of the Transaction Journal records that are created.	Small, 54.0	54.0

## Points Change Output

Output representation of the points debited from or credited to the member's points balance by the transaction journal.

Property Name	Type	Description	Filter Group and Version	Available Version
changeInPoints	String	Specifies the number of points that have been debited from or credited to the member by the transaction journal.	Small, 58.0	58.0
<del>loyaltyMemberCurrency</del>	String	Name of the loyalty member currency for which the transaction journal has debited or credited points.	Small, 58.0	58.0

## Points Summary

Output representation of the point summary.

Property Name	Type	Description	Filter Group and Version	Available Version
<del>changeInEscrowPoints</del> Balance	Double	The change in the escrow points balance of the loyalty program member associated with the transaction journal.	Small, 55.0	55.0
<del>changeInPointsBalance</del>	Double	The change in the points balance of the loyalty program member associated with the transaction journal.	Small, 55.0	55.0
<del>loyaltyProgramCurrency</del> Name	String	The name of the loyalty program currency for which points are debited from or credited to the loyalty program member associated with the transaction journal.	Small, 55.0	55.0
<del>loyaltyProgramCurrency</del> Type	CurrencyType	The type of the loyalty program currency. Valid values are: <ul style="list-style-type: none"> <li>NonQualifying</li> <li>Qualifying</li> </ul>	Small, 55.0	55.0

## Promotion Details

Output representation of the details of an eligible promotion.

Property Name	Type	Description	Filter Group and Version	Available Version
additionalPromotionFields	Map<String, Object>	Custom fields, and the fields in the context service definition that are mapped to the fields in the promotion. These fields represent the additional details of the promotion.	Small, 60.0	60.0
currencyIsoCode	String	Three-letter ISO 4217 code of the active monetary currency associated with the promotion.	Small, 60.0	60.0
displayName	String	Name of the eligible promotion.	Small, 60.0	60.0
priorityNumber	Integer	Number that determines the priority of the promotion in a list of eligible promotions.  Multiple promotions can have the same priority number.	Small, 60.0	60.0
promotionCode	String	Code of the eligible promotion.	Small, 60.0	60.0
promotionEligibleRules	<a href="#">Promotion Rules List</a> []	List of eligible promotion rules for the customer's cart.	Small, 60.0	60.0
promotionId	String	ID of the eligible promotion.	Small, 60.0	60.0
promotionLimits	<a href="#">Promotion Limit</a> []	List of promotion limits on the cart and the cart items. The list contains the details of how much of the limit has already been attained so far.	Small, 60.0	60.0
couponDetails	<a href="#">Promotion Coupon Availability</a> on page 440	The coupon that is eligible for the customers' cart.	Small, 60.0	60.0
isAutomatic	Boolean	Indicates whether the promotion is automatically applied for eligible carts (true) or not (false). The default value is false	Small, 60.0	60.0

## Promotion Limit

Output representation of the promotion limit of an eligible promotion.

Property Name	Type	Description	Filter Group and Version	Available Version
limitCurrentAttainment	Double	Value of the promotion's limit that has been attained so far.	Small, 60.0	60.0
limitTarget	Double	Value beyond which the promotion can't be applied for customer carts.	Small, 60.0	60.0

Property Name	Type	Description	Filter Group and Version	Available Version
limitType	String	Type of the promotion limit. Possible values are: <ul style="list-style-type: none"> <li>CartItemCount—Maximum number of cart items the promotion applies to.</li> <li>OrderCount—Maximum number of carts the promotion applies to.</li> </ul>	Small, 60.0	60.0

## Promotion Limit Output

Output representation of promotion limits.

Property Name	Type	Description	Filter Group and Version	Available Version
liabilityLimit	Double	Maximum financial liability the promotion can incur.	Small, 60.0	60.0
liabilityPerCustomerLimit	Double	Maximum financial liability the promotion can incur for a customer.	Small, 60.0	60.0
loyaltyProgramCurrency	String	Active loyalty program currency associated with the loyalty points.	Small, 60.0	60.0
perCartLimit	Integer	Maximum number of carts that the promotion can be applied for.	Small, 60.0	60.0
perLineItemLimit	Integer	Maximum number of cart items that the promotion can be applied for.	Small, 60.0	60.0
pointsLimit	Double	Maximum number of points the promotion can credit.	Small, 60.0	60.0
pointsPerCustomerLimit	Integer	Maximum number of points the promotion can credit to a customer.	Small, 60.0	60.0
usePerCustomerLimit	Integer	Maximum number of times the promotion can be applied for a customer.	Small, 60.0	60.0
vouchersLimit	Integer	Maximum number of vouchers the promotion can issue.	Small, 60.0	60.0
vouchersPerCustomerLimit	Integer	Maximum number of vouchers the promotion can issue to a customer.	Small, 60.0	60.0

## Promotion Reward Details

Output representation of the details of the reward offered by an eligible promotion rule.

Property Name	Type	Description	Filter Group and Version	Available Version
rewardDetails	Map<String, Object>	Details of the reward offered by the promotion rule.	Small, 60.0	60.0
rewardType	String	Type of the reward offered by the promotion rule. Possible values are: <ul style="list-style-type: none"> <li>• <code>AssignBadge</code>—Assigns a badge to the loyalty program member.</li> <li>• <code>AssignGame</code>—Enrolls the customer in a game, playing which the customer can win rewards.</li> <li>• <code>CreditFixedPoints</code>—Credits a fixed number of loyalty points to the loyalty program member.</li> <li>• <code>CreditMultiplierPoints</code>—Credits loyalty points that are based on the multiplier and transaction journal fields to the loyalty program member.</li> <li>• <code>GiveFreeProduct</code>—provides the customer a free product on the cart or cart items.</li> <li>• <code>IssueVoucher</code>—Issues a voucher to the customer.</li> <li>• <code>ProvideDiscount</code>—Provides the customer a discount on the cart or cart items.</li> </ul>	Small, 60.0	60.0

## Promotion Rules List

Output representation of an eligible promotion rule.

Property Name	Type	Description	Filter Group and Version	Available Version
ruleName	String	Name of the eligible promotion rule.	Small, 60.0	60.0
ruleRewards	<a href="#">Promotion Reward Details</a> []	Rewards associated with the eligible rule.	Small, 60.0	60.0

## Redeem Voucher Output

Represents the response for the redeem voucher POST method.

**Sample Response: Reserve**

```
{
  "reservationKey": "key1",
  "reservedValue": 16.34,
  "status": "Reserved"
  "currencyIsoCode": "EUR"
}
```

**Sample Response: Reinstate - Reserved to Issued**


```
{
  "remainingAmount": 98.32,
  "status": "Issued"
}
```

**Sample Response: Reinstate - Reserved to Expired**

```
{
  "status": "Expired"
}
```

**Sample Response: Redeem**

```
{
  "remainingAmount": 98.32,
  "status": "Redeemed",
  "currencyIsoCode": "EUR"
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
remainingAmount	Double	The voucher value after redemption.  <b>Note:</b> If the complete value of the voucher is redeemed, the remaining amount is zero.	Small, 53.0	53.0
currencyIsoCode	String	The ISO code of the currency.	Small, 53.0	53.0
status	VoucherStatus (Enum)	Specifies the status of the voucher.	Small, 62.0	62.0
reservationKey	String	Specifies the key used to redeem or reinstate a reserved voucher.	Small, 62.0	62.0
reservedValue	Currency	Specifies the amount reserved for partially redeemable vouchers.	Small, 62.0	62.0

**Summary**

Output representation of the execution summary of the transaction journals.

Property Name	Type	Description	Filter Group and Version	Available Version
pointsSummary	<a href="#">Points Summary[]</a>	A summary of points that are debited from or credited to the points balance or the escrow points balance of the loyalty program member associated with the transaction journal.	Small, 55.0	55.0
vouchersIssuedCount	Integer	The number of vouchers that are issued to the loyalty program member associated with the transaction journal.	Small, 55.0	55.0

## Transaction Details Output

Output representation of the transaction journal ledger records.

Property Name	Type	Description	Filter Group and Version	Available Version
activityDate	String	The activity date of the transaction journal.	Small, 58.0	58.0
<del>externalTransactionNumber</del>	String	The unique transaction number from an external system.	Small, 58.0	58.0
journalSubTypeName	String	The name of the journal subtype associated with the transaction journal record.	Small, 58.0	58.0
journalTypeName	String	The name of the journal type associated with the transaction journal record.	Small, 58.0	58.0
pointsChange	<a href="#">Points Change Output</a>	The details of the points debited from or credited to the member's points balance by the transaction journal.	Small, 58.0	58.0
<del>productCategoryName</del>	String	The name of the product category associated with the transaction journal.	Small, 58.0	58.0
productName	String	The name of the product associated with the transaction journal.	Small, 58.0	58.0
transactionAmount	String	The amount that the member spent on the transaction journal.	Small, 58.0	58.0
transactionJournalId	String	The ID of the transaction journal record.	Small, 58.0	58.0
<del>transactionJournalNumber</del>	String	The unique number of the transaction journal record.	Small, 58.0	58.0

## Transaction Journal Output

Output representation of the transaction journals.



Property Name	Type	Description	Filter Group and Version	Available Version
activityDate	String	The date and time of when transaction journal is created.	Small, 55.0	55.0
journalSubType	String	The journal subtype of the transaction journal.	Small, 55.0	55.0
journalType	String	The journal type of the transaction journal.	Small, 55.0	55.0
loyaltyProgram	String	The loyalty program related with the transaction journal.	Small, 55.0	55.0
loyaltyProgramMember	String	The membership number of the loyalty program related to the transaction journal.	Small, 55.0	55.0
referredMember	String	The membership number of the loyalty program member who has been referred by the member.	Small, 55.0	55.0
status	String	The status of the transaction journal.	Small, 55.0	55.0
transactionJournalId	String	The ID of the transaction journal that is created.	Small, 55.0	55.0

## Transaction Journal Map Output

Output representation of the loyalty program process simulation result.

Property Name	Type	Description	Filter Group and Version	Available Version
transactionJournals	<a href="#">Transaction Journal Result[]</a>	Represents the transaction journals on which the simulation was performed.	Small, 55.0	55.0

## Transaction Journal Result

Output representation of the transaction journals result.

Property Name	Type	Description	Filter Group and Version	Available Version
errorMessage	String	Specifies the error message if the process has no rules or if no process qualifies for the transaction journal.	Small, 55.0	55.0
executionSummary	<a href="#">Summary</a>	Represents the execution summary of the transaction journals.	Small, 55.0	55.0
processId	String	The ID of the loyalty program process used to process the transaction journals.	Small, 55.0	55.0

Property Name	Type	Description	Filter Group and Version	Available Version
processName	String	The name of the loyalty program process used to process the transaction journals.	Small, 55.0	55.0
processRules	<a href="#">Loyalty Program Process Rule[]</a>	The list of rules defined in the loyalty program process used to process the transaction journals.	Small, 55.0	55.0

## Transaction Journal Process Result

Output representation of the Transaction Journal result.

Property Name	Type	Description	Filter Group and Version	Available Version
badgesAssigned	<a href="#">Badge Output[]</a>	List of badges that are assigned to the member.	Small, 60.0	60.0
changeInMemberPoints	<a href="#">Member Points Balance Change[]</a>	Details of points that are credited to or debited from the member's points balance and escrow points balance for all the currencies associated with the member.	Small, 54.0	54.0
gamesAssigned	<a href="#">Game Assignment Output[]</a>	List of games that are assigned to the member.	Small, 60.0	60.0
id	String	Details of the Transaction Journal records that are created.	Small, 54.0	54.0
voucherResult	<a href="#">Voucher Result</a>	List of vouchers that are issued or redeemed.	Small, 54.0	54.0

## Transaction Ledgers Output

Output representation of the transaction journals and the ledgers associated with the specified loyalty program member.

### Sample Response

```
{
  "message": null,
  "status": true,
  "transactionJournalCount": 2,
  "externalTransactionNumber" : "123123"
  "transactionJournals": [
    {
      "activityDate": "2023-03-01T08:00:00.000Z",
      "additionalAttributes": [
        {
          "fieldName": "additionalInfo__c",
          "value": "null"
        },
        {
```

```

        "fieldName": "Destination",
        "value": "null"
      }
    ],
    "journalSubTypeName": "Product Buy",
    "journalTypeName": "Accrual",
    "pointsChange": [],
    "transactionJournalId": "01Vxx00000002rREAQ"
  },
  {
    "activityDate": "2023-02-01T20:00:00.000Z",
    "additionalAttributes": [
      {
        "fieldName": "additionalInfo__c",
        "value": "test123"
      },
      {
        "fieldName": "Destination",
        "value": "Jaipur"
      }
    ],
    "journalSubTypeName": "Member Enrollment",
    "journalTypeName": "Accrual",
    "pointsChange": [
      {
        "changeInPoints": "-100.0",
        "loyaltyMemberCurrency": "Tier Points"
      },
      {
        "changeInPoints": "90.0",
        "loyaltyMemberCurrency": "Reward Points"
      }
    ],
    "productCategoryName": "Gift",
    "productName": "New Member Gift Package",
    "transactionAmount": "1000.0",
    "transactionJournalId": "01Vxx00000000H1EAI"
  }
]
}

```

Property Name	Type	Description	Filter Group and Version	Available Version
message	String	The request response message.	Small, 58.0	58.0
status	Boolean	Indicates whether the request was successful ( <code>true</code> ) or not ( <code>false</code> ).	Small, 58.0	58.0
transactionJournalCount	Integer	The number of transaction journal records created for the member.	Small, 58.0	58.0
transactionJournals	<a href="#">Transaction Details Output</a>	The details of the transaction journals.	Small, 58.0	58.0

## Update Cumulative Usage Action Output

Output representation of the update cumulative usage action type.

Property Name	Type	Description	Filter Group and Version	Available Version
actionType	String	The type of action in the loyalty program process rule.	Small, 55.0	55.0
promotion	String	The name of promotion for which the member's cumulative usage is updated or reset.	Small, 55.0	55.0
promotionId	String	The ID of the promotion for which the member's cumulative usage is updated or reset.	Small, 55.0	55.0
stepName	String	The name of the step in the loyalty program process rule.	Small, 55.0	55.0
updatedCumulativePromotionUsage	Integer	The updated value of the loyalty program member's usage towards the cumulative promotion.	Small, 55.0	55.0

## Voucher

Output representation of the details of a voucher.

Property Name	Type	Description	Filter Group and Version	Available Version
currencyIsoCode	String	The currency ISO code in which the member has redeemed the voucher.	Small, 58.0	58.0
description	String	The description of the voucher issued to the member.	Small, 58.0	58.0
discountPercent	Integer	The discount percentage that's offered to the member by the voucher.	Small, 58.0	58.0
effectiveDate	String	The date the member can redeem the voucher.	Small, 58.0	58.0
effectiveDateTime	DateTime	The date and time from when the voucher is available for redemption.	Small, 62.0	62.0
expirationDate	String	The date the voucher expires.	Small, 58.0	58.0
expirationDateTime	DateTime	The date and time when the voucher expires and is no longer available for redemption.	Small, 62.0	62.0

Property Name	Type	Description	Filter Group and Version	Available Version
faceValue	Double	The value that the member can redeem using the voucher.	Small, 58.0	58.0
isVoucherDefinitionActive	Boolean	Indicates whether the voucher definition associated with the voucher is active ( <code>true</code> ) or not ( <code>false</code> ).	Small, 58.0	58.0
isVoucherPartiallyRedeemable	Boolean	Indicates whether the voucher can be redeemed across multiple transactions ( <code>true</code> ) or only one transaction ( <code>false</code> ).	Small, 58.0	58.0
hasTimeBasedVoucher	Boolean	Indicates whether the voucher has an effective date and time ( <code>true</code> ) or not ( <code>false</code> ).	Small, 62.0	62.0
partnerAccountName	String	The name of the partner associated with the voucher's voucher definition.	Small, 58.0	58.0
productCategoryId	String	The ID of product category associated with the voucher's voucher definition.	Small, 58.0	58.0
productCategoryName	String	The name of the product category associated with the voucher definition.	Small, 58.0	58.0
productId	String	The ID of the product associated with the voucher definition.	Small, 58.0	58.0
productName	String	The name of the product associated with the voucher definition.	Small, 58.0	58.0
promotionId	String	The ID of the promotion for which the voucher is issued.	Small, 58.0	58.0
promotionName	String	The name of the promotion for which the voucher is issued.	Small, 58.0	58.0
redeemedValue	Double	The value that the member has already redeemed from a partially redeemable voucher.	Small, 58.0	58.0
remainingValue	Double	The remaining value of the voucher after partial redemption.	Small, 58.0	58.0
status	String	The status of the voucher.	Small, 58.0	58.0
type	String	The type of voucher issued to the member.	Small, 58.0	58.0
useDate	String	The date the voucher was redeemed.	Small, 58.0	58.0
voucherCode	String	The code of the voucher.	Small, 58.0	58.0
voucherDefinition	String	The name of the voucher definition used to issue the voucher to the member.	Small, 58.0	58.0
voucherId	String	The ID of the voucher record.	Small, 58.0	58.0

Property Name	Type	Description	Filter Group and Version	Available Version
voucherImageUrl	String	The URL of the image associated with the voucher definition.	Small, 58.0	58.0
voucherNumber	String	The unique number of the voucher.	Small, 58.0	58.0

## Vouchers List

Output representation of the list of vouchers associated with the loyalty program member.

### JSON example

```
{
  "voucherCount": 6,
  "vouchers": [
    {
      "effectiveDate": "2020-11-01",
      "effectiveDateTime": "2020-11-01T07:47:44.000Z",
      "expirationDate": "2021-11-01",
      "expirationDateTime": "2021-11-01T07:33:00.000Z",
      "faceValue": 100,
      "isVoucherDefinitionActive": true,
      "isVoucherPartiallyRedeemable": false,
      "hasTimeBasedVoucherPeriod": true,
      "redeemedValue": 0,
      "remainingValue": 100,
      "status": "Expired",
      "type": "FixedValue",
      "voucherCode": "J4XN26MT",
      "voucherDefinition": "Ajio",
      "voucherId": "0kDT1000000TUwsMAG",
      "voucherNumber": "00005852"
    },
    {
      "effectiveDate": "2020-11-01",
      "expirationDate": "2021-11-01",
      "faceValue": 100,
      "isVoucherDefinitionActive": true,
      "isVoucherPartiallyRedeemable": false,
      "hasTimeBasedVoucherPeriod": false,
      "redeemedValue": 0,
      "remainingValue": 100,
      "status": "Expired",
      "type": "FixedValue",
      "voucherCode": "6PM01LT8",
      "voucherDefinition": "Ajio",
      "voucherId": "0kDT1000000TUjEMAW",
      "voucherNumber": "00005006"
    },
    {
      "effectiveDate": "2020-11-01",
      "expirationDate": "2021-11-01",

```

```
    "faceValue": 100,
    "isVoucherDefinitionActive": true,
    "isVoucherPartiallyRedeemable": false,
    "hasTimeBasedVoucherPeriod": false,
    "redeemedValue": 0,
    "remainingValue": 100,
    "status": "Expired",
    "type": "FixedValue",
    "voucherCode": "LEEFAQGE",
    "voucherDefinition": "Ajio",
    "voucherId": "0kDT1000000TUQcMAO",
    "voucherNumber": "00003852"
  },
  {
    "effectiveDate": "2020-11-01",
    "expirationDate": "2021-11-01",
    "faceValue": 100,
    "isVoucherDefinitionActive": true,
    "isVoucherPartiallyRedeemable": false,
    "hasTimeBasedVoucherPeriod": false,
    "redeemedValue": 0,
    "remainingValue": 100,
    "status": "Expired",
    "type": "FixedValue",
    "voucherCode": "9WOYZK2G",
    "voucherDefinition": "Ajio",
    "voucherId": "0kDT1000000TUCyMAO",
    "voucherNumber": "00003006"
  },
  {
    "effectiveDate": "2020-11-01",
    "expirationDate": "2021-11-01",
    "faceValue": 100,
    "isVoucherDefinitionActive": true,
    "isVoucherPartiallyRedeemable": false,
    "hasTimeBasedVoucherPeriod": false,
    "redeemedValue": 0,
    "remainingValue": 100,
    "status": "Expired",
    "type": "FixedValue",
    "voucherCode": "6PLO64G1",
    "voucherDefinition": "Ajio",
    "voucherId": "0kDT1000000TU6yMAG",
    "voucherNumber": "00002634"
  },
  {
    "effectiveDate": "2020-11-01",
    "expirationDate": "2021-11-01",
    "faceValue": 100,
    "isVoucherDefinitionActive": true,
    "isVoucherPartiallyRedeemable": false,
    "hasTimeBasedVoucherPeriod": false,
    "redeemedValue": 0,
    "remainingValue": 100,
```

```


    "status": "Expired",
    "type": "FixedValue",
    "voucherCode": "K54Y02BK",
    "voucherDefinition": "Ajio",
    "voucherId": "0kDT1000000TTuKMAW",
    "voucherNumber": "00001850"
  }
]
}

```

Property Name	Type	Description	Filter Group and Version	Available Version
voucherCount	Integer	The number of vouchers returned by the API request.	Small, 58.0	58.0
vouchers	<a href="#">Voucher</a>	The details of the vouchers.	Small, 58.0	58.0

## Voucher Result

Output representation of the voucher result.

Property Name	Type	Description	Filter Group and Version	Available Version
vouchersIssued	<a href="#">Voucher Output[]</a>	List of the vouchers that are issued to the associated loyalty program member.	Small, 54.0	54.0
		 <b>Note:</b> This output is provided when the process contains an Issue Voucher action.		

## Voucher Output

Output representation of the details about a voucher.

Property Name	Type	Description	Filter Group and Version	Available Version
discountPercent	Integer	The discount percentage associated with the voucher.	Small, 54.0	54.0
effectiveDate	String	The date the voucher goes into effect.	Small, 54.0	54.0
expirationDate	String	The date the voucher expires.	Small, 54.0	54.0
faceValue	Double	The value of the voucher.	Small, 54.0	54.0
productId	String	The ID of the product in the voucher issued.	Small, 54.0	54.0
promotionId	String	The ID of the promotion associated with the voucher.	Small, 54.0	54.0



Property Name	Type	Description	Filter Group and Version	Available Version
voucherCode	String	The voucher code.	Small, 54.0	54.0
voucherId	String	The ID of the voucher issued.	Small, 54.0	54.0

## Promotion Configuration Output

Output representation of the promotion's details and its configuration.

### JSON Example

```
{
  "apiExecutedDateTime": "2024-05-29T09:02:38.809Z",
  "promotionDetails": {
    "additionalFieldValues": {
      "attributes": {}
    },
    "displayName": "Tea Time",
    "endDateTime": "2024-07-31T08:58:00.000Z",
    "name": "Tea Time",
    "priorityNumber": 1,
    "promotionCode": "TEATIME",
    "promotionLimits": {},
    "startDateTime": "2024-05-01T08:58:00.000Z"
  },
  "promotionLastModifiedDateTime": "2024-05-29T09:01:51.000Z",
  "promotionRuleList": [
    {
      "apiName": "Eligibility_Rule_Unified_Promotions_0c8xx000000034L",
      "isEligibilityRule": true,
      "name": "Eligibility_Rule_Unified_Promotions_0c8xx000000034L_v1",
      "ruleActionList": [],
      "ruleFilterCriteriaList": [
        {
          "conditionLogic": "1",
          "name": "FilterCriteria1",
          "ruleActionList": [],
          "ruleConditionList": [
            {
              "operator": "Equals",
              "parameterName": "CurrencyISOCode",
              "sequenceNumber": 1,
              "valueList": [
                "USD"
              ],
              "valueType": "Literal"
            }
          ],
          "sequenceNumber": 1,
          "type": "Exists"
        }
      ],
      "sequenceNumber": 1,
      "type": "Exists"
    }
  ]
}
```

```

"conditionLogic": "1 AND 2",
"name": "FilterCriteria2",
"parentFilterCriteria": "FilterCriteria1",
"ruleActionList": [],
"ruleConditionList": [
  {
    "operator": "GreaterThanOrEquals",
    "parameterName": "ActivityStartDateTime",
    "sequenceNumber": 1,
    "valueList": [
      "2024-05-01 08:58:00"
    ],
    "valueType": "Literal"
  },
  {
    "operator": "LessThanOrEquals",
    "parameterName": "ActivityStartDateTime",
    "sequenceNumber": 2,
    "valueList": [
      "2024-07-31 08:58:00"
    ],
    "valueType": "Literal"
  }
],
"sequenceNumber": 2,
"type": "Exists"
},
{
"conditionLogic": "1 AND 2",
"name": "FilterCriteria3",
"parentFilterCriteria": "FilterCriteria2",
"ruleActionList": [],
"ruleConditionList": [
  {
    "operator": "Equals",
    "parameterName": "ProductCatalog",
    "sequenceNumber": 1,
    "valueList": [
      "Beverages"
    ],
    "valueType": "Literal"
  },
  {
    "operator": "In",
    "parameterName": "EligibleCartLineProduct",
    "sequenceNumber": 2,
    "valueList": [
      "Tea"
    ],
    "valueType": "Literal"
  }
],
"sequenceNumber": 3,
"type": "Exists"
}

```

```
    }
  ],
  "ruleParameterList": [
    {
      "contextTag": "cartLineProductCatalogId",
      "dataType": "Text",
      "decimalPlaces": 0,
      "hasContextTagMapping": true,
      "isCollection": true,
      "isInput": true,
      "isOutput": false,
      "name": "ProductCatalog",
      "type": "Tag"
    },
    {
      "contextTag": "activityStartDate",
      "dataType": "DateTime",
      "decimalPlaces": 0,
      "hasContextTagMapping": true,
      "isCollection": true,
      "isInput": true,
      "isOutput": false,
      "name": "ActivityStartDateTime",
      "type": "Tag"
    },
    {
      "contextTag": "currencyISOCODE",
      "dataType": "Text",
      "decimalPlaces": 0,
      "hasContextTagMapping": true,
      "isCollection": true,
      "isInput": true,
      "isOutput": false,
      "name": "CurrencyISOCODE",
      "type": "Tag"
    },
    {
      "contextTag": "cartLineProductId",
      "dataType": "Text",
      "decimalPlaces": 0,
      "hasContextTagMapping": true,
      "isCollection": true,
      "isInput": true,
      "isOutput": false,
      "name": "EligibleCartLineProduct",
      "type": "Tag"
    }
  ],
  "sequenceNumber": 2147483647
},
{
  "apiName": "DiscountTea10Unified_Promotions_0c8xx000000034L",
  "isEligibilityRule": false,
  "name": "DiscountTea10Unified_Promotions_0c8xx000000034L_v1",
```

```

"ruleActionList": [],
"ruleFilterCriteriaList": [
  {
    "conditionLogic": "1",
    "name": "FinalFilterCriteria",
    "ruleActionList": [],
    "ruleConditionList": [
      {
        "operator": "GreaterThanOrEquals",
        "parameterName": "TransactionAmount",
        "sequenceNumber": 1,
        "valueList": [
          "25.0"
        ],
        "valueType": "Literal"
      }
    ],
    "sequenceNumber": 1,
    "type": "Exists"
  },
  {
    "conditionLogic": "1",
    "name": "NonLoyaltyMemberCriteria",
    "parentFilterCriteria": "BranchCriteria",
    "ruleActionList": [
      {
        "actionType": "SetValue",
        "name": "pricingReward1",
        "ruleActionParameterList": [
          {
            "name": "type",
            "sequenceNumber": 1,
            "value": "ProvideDiscount",
            "valueType": "Literal"
          },
          {
            "name": "discountLevel",
            "sequenceNumber": 2,
            "value": "Cart",
            "valueType": "Literal"
          },
          {
            "name": "discountValue",
            "sequenceNumber": 3,
            "value": "10.0",
            "valueType": "Literal"
          },
          {
            "name": "discountType",
            "sequenceNumber": 4,
            "value": "PercentageOff",
            "valueType": "Literal"
          }
        ]
      }
    ],
  }
]

```

```

        "sequenceNumber": 1
      }
    ],
    "ruleConditionList": [
      {
        "operator": "GreaterThanOrEquals",
        "parameterName": "ActivityStartDateTime",
        "sequenceNumber": 1,
        "valueList": [
          "1970-01-01 08:00:00"
        ],
        "valueType": "Literal"
      }
    ],
    "sequenceNumber": 3,
    "type": "Exists"
  },
  {
    "name": "BranchCriteria",
    "parentFilterCriteria": "FinalFilterCriteria",
    "ruleActionList": [],
    "ruleConditionList": [],
    "sequenceNumber": 2,
    "type": "Branch"
  }
],
"ruleParameterList": [
  {
    "contextTag": "transactionAmount",
    "dataType": "Currency",
    "decimalPlaces": 0,
    "hasContextTagMapping": true,
    "isCollection": true,
    "isInput": true,
    "isOutput": false,
    "name": "TransactionAmount",
    "type": "Tag"
  },
  {
    "dataType": "Text",
    "decimalPlaces": 0,
    "hasContextTagMapping": false,
    "isCollection": false,
    "isInput": false,
    "isOutput": true,
    "name": "discountLevel",
    "type": "Variable"
  },
  {
    "dataType": "Text",
    "decimalPlaces": 0,
    "hasContextTagMapping": false,
    "isCollection": false,
    "isInput": false,

```

```

        "isOutput": true,
        "name": "discountType",
        "type": "Variable"
    },
    {
        "contextTag": "activityStartDate",
        "dataType": "DateTime",
        "decimalPlaces": 0,
        "hasContextTagMapping": true,
        "isCollection": true,
        "isInput": true,
        "isOutput": false,
        "name": "ActivityStartDateTime",
        "type": "Tag"
    },
    {
        "dataType": "Text",
        "decimalPlaces": 0,
        "hasContextTagMapping": false,
        "isCollection": false,
        "isInput": false,
        "isOutput": true,
        "name": "type",
        "type": "Variable"
    },
    {
        "dataType": "Numeric",
        "decimalPlaces": 0,
        "hasContextTagMapping": false,
        "isCollection": false,
        "isInput": false,
        "isOutput": true,
        "name": "discountValue",
        "type": "Variable"
    }
],
"sequenceNumber": 1
},
{
    "apiName": "TeaAndCoffeeDiscount10Unified_Promotions_0c8xx000000034L",
    "isEligibilityRule": false,
    "name": "TeaAndCoffeeDiscount10Unified_Promotions_0c8xx000000034L_v1",
    "ruleActionList": [],
    "ruleFilterCriteriaList": [
        {
            "conditionLogic": "1",
            "name": "NonLoyaltyMemberCriteria",
            "parentFilterCriteria": "BranchCriteria",
            "ruleActionList": [
                {
                    "actionType": "SetValue",
                    "name": "pricingLineLevelReward1",
                    "ruleActionParameterList": [
                        {

```

```

        "name": "type",
        "sequenceNumber": 1,
        "value": "ProvideDiscount",
        "valueType": "Literal"
    },
    {
        "name": "discountType",
        "sequenceNumber": 2,
        "value": "PercentageOff",
        "valueType": "Literal"
    },
    {
        "name": "discountLevel",
        "sequenceNumber": 3,
        "value": "Line",
        "valueType": "Literal"
    },
    {
        "name": "discountValue",
        "sequenceNumber": 4,
        "value": "10.0",
        "valueType": "Literal"
    },
    {
        "name": "discountProduct",
        "sequenceNumber": 5,
        "value": "Tea",
        "valueType": "Literal"
    },
    {
        "name": "discountProductId",
        "sequenceNumber": 6,
        "value": "Tea",
        "valueType": "Literal"
    },
    {
        "name": "discountAppliedAt",
        "sequenceNumber": 7,
        "value": "LineItem",
        "valueType": "Literal"
    }
    ],
    "sequenceNumber": 1
},
{
    "actionType": "SetValue",
    "name": "pricingLineLevelReward2",
    "ruleActionParameterList": [
        {
            "name": "type",
            "sequenceNumber": 1,
            "value": "ProvideDiscount",
            "valueType": "Literal"
        },
    ],

```

```

        {
            "name": "discountType",
            "sequenceNumber": 2,
            "value": "PercentageOff",
            "valueType": "Literal"
        },
        {
            "name": "discountLevel",
            "sequenceNumber": 3,
            "value": "Line",
            "valueType": "Literal"
        },
        {
            "name": "discountValue",
            "sequenceNumber": 4,
            "value": "10.0",
            "valueType": "Literal"
        },
        {
            "name": "discountProduct",
            "sequenceNumber": 5,
            "value": "Coffee",
            "valueType": "Literal"
        },
        {
            "name": "discountProductId",
            "sequenceNumber": 6,
            "value": "Coffee",
            "valueType": "Literal"
        },
        {
            "name": "discountAppliedAt",
            "sequenceNumber": 7,
            "value": "LineItem",
            "valueType": "Literal"
        }
    ],
    "sequenceNumber": 2
}
],
"ruleConditionList": [
    {
        "operator": "GreaterThanOrEquals",
        "parameterName": "ActivityStartDateTime",
        "sequenceNumber": 1,
        "valueList": [
            "1970-01-01 08:00:00"
        ],
        "valueType": "Literal"
    }
],
"sequenceNumber": 4,
"type": "Exists"
},

```



```

    {
      "conditionLogic": "1",
      "name": "ProductFilter1",
      "ruleActionList": [],
      "ruleConditionList": [
        {
          "operator": "In",
          "parameterName": "CartLineProduct",
          "sequenceNumber": 1,
          "valueList": [
            "Tea"
          ],
          "valueType": "Literal"
        }
      ],
      "sequenceNumber": 1,
      "type": "Loop"
    },
    {
      "name": "BranchCriteria",
      "parentFilterCriteria": "FinalFilterCriteria",
      "ruleActionList": [],
      "ruleConditionList": [],
      "sequenceNumber": 3,
      "type": "Branch"
    },
    {
      "conditionLogic": "1",
      "name": "FinalFilterCriteria",
      "ruleActionList": [],
      "ruleConditionList": [
        {
          "matchType": "Any",
          "operator": "GreaterThanOrEquals",
          "parameterName": "FilterProducts1",
          "sequenceNumber": 1,
          "valueList": [
            "6.0"
          ],
          "valueType": "Literal"
        }
      ],
      "sequenceNumber": 2,
      "type": "Exists"
    }
  ],
  "ruleParameterList": [
    {
      "dataType": "Text",
      "decimalPlaces": 0,
      "hasContextTagMapping": false,
      "isCollection": false,
      "isInput": false,
      "isOutput": true,
    }
  ]
}

```

```

        "name": "discountLevel",
        "type": "Variable"
    },
    {
        "dataType": "Text",
        "decimalPlaces": 0,
        "hasContextTagMapping": false,
        "isCollection": false,
        "isInput": false,
        "isOutput": true,
        "name": "discountProduct",
        "type": "Variable"
    },
    {
        "dataType": "Text",
        "decimalPlaces": 0,
        "hasContextTagMapping": false,
        "isCollection": false,
        "isInput": false,
        "isOutput": true,
        "name": "discountProductId",
        "type": "Variable"
    },
    {
        "dataType": "Text",
        "decimalPlaces": 0,
        "hasContextTagMapping": false,
        "isCollection": false,
        "isInput": false,
        "isOutput": true,
        "name": "discountType",
        "type": "Variable"
    },
    {
        "contextTag": "activityStartDate",
        "dataType": "DateTime",
        "decimalPlaces": 0,
        "hasContextTagMapping": true,
        "isCollection": true,
        "isInput": true,
        "isOutput": false,
        "name": "ActivityStartDateTime",
        "type": "Tag"
    },
    {
        "contextTag": "cartLineProductId",
        "dataType": "Text",
        "decimalPlaces": 0,
        "hasContextTagMapping": true,
        "isCollection": true,
        "isInput": true,
        "isOutput": false,
        "name": "CartLineProduct",
        "type": "Tag"
    }

```

```

    },
    {
      "dataType": "Text",
      "decimalPlaces": 0,
      "hasContextTagMapping": false,
      "isCollection": false,
      "isInput": false,
      "isOutput": true,
      "name": "type",
      "type": "Variable"
    },
    {
      "dataType": "Numeric",
      "decimalPlaces": 0,
      "hasContextTagMapping": false,
      "isCollection": false,
      "isInput": false,
      "isOutput": true,
      "name": "discountValue",
      "type": "Variable"
    },
    {
      "aggregateFunction": "Sum",
      "dataType": "Tuple",
      "decimalPlaces": 0,
      "groupByContextTag": "cartLineProductId",
      "hasContextTagMapping": true,
      "isCollection": true,
      "isInput": true,
      "isOutput": true,
      "name": "FilterProducts1",
      "relatedFilterCriteriaName": "ProductFilter1",
      "type": "Accumulate",
      "value": "cartLineItemQuantity"
    },
    {
      "dataType": "Text",
      "decimalPlaces": 0,
      "hasContextTagMapping": false,
      "isCollection": false,
      "isInput": false,
      "isOutput": true,
      "name": "discountAppliedAt",
      "type": "Variable"
    }
  ],
  "sequenceNumber": 2
}
],
"relatedRuleLibrary": {
  "apiName": "GPMRuleLibrary",
  "description": "GPMRuleLibrary",
  "name": "GPMRuleLibrary_v1",
  "relatedContextDefinitionDeveloperName": "GPMContext"
}

```

```

    }
  }
}

```

**Table 1: Promotion Configuration Output**

Property Name	Type	Description	Filter Group and Version	Available Version
apiExecutedDateTime	String	The date and time on which the API was executed.	Small, 61.0	61.0
promotionDetails	Promotion Details []	The details of the promotion.	Small, 61.0	61.0
promotionLastModifiedDate	String	The date and time on which the promotion was last modified.	Small, 61.0	61.0
promotionRuleList	Promotion Rule List []	The rules of the promotion. The rules are represented as context rules in the response.	Small, 61.0	61.0
relatedRuleLibrary	Rule Library []	The rule library that's associated with the promotion.	Small, 61.0	61.0

**Table 2: Promotion Details Output**

Property Name	Type	Description	Filter Group and Version	Available Version
name	String	The name of the promotion.	Small, 61.0	61.0
objective	String	The objective of the promotion.	Small, 61.0	61.0
priorityNumber	String	The priority number of the promotion.	Small, 61.0	61.0
startDateTime	String	The date and time from when the promotion can be applied for customers.	Small, 61.0	61.0
endDateTime	String	The date and time until the promotion can be applied for customers.	Small, 61.0	61.0
description	String	The description of the promotion.	Small, 61.0	61.0
additionalFieldValues	Additonal Attributes []	The details of the promotion's additional fields.	Small, 61.0	61.0
promotionCode	String	The code of the promotion.	Small, 61.0	61.0
currencyIsoCode	String	Three-letter ISO 4217 code currency associated with the promotion.	Small, 61.0	61.0

**Table 3: Promotion Additional Attributes Output**

Property Name	Type	Description	Filter Group and Version	Available Version
attributes	Map<String,String>	The additional attributes of the promotion.	Small, 61.0	61.0

**Table 4: Rule Library Output**

Property Name	Type	Description	Filter Group and Version	Available Version
name	String	The name of the rule library associated with the promotion.	Small, 61.0	61.0
apiName	String	The API name of the rule library associated with the promotion.	Small, 61.0	61.0
description	String	The description of the rule library associated with the promotion.	Small, 61.0	61.0
<del>developerName</del> contextName	String	The developer name of the context definition associated with the promotion's rule library.	Small, 61.0	61.0

**Table 5: Promotion Rule Output**

Property Name	Type	Description	Filter Group and Version	Available Version
name	String	The name of the promotion rule. A promotion consists of two types of rules. The eligibility rule decides which products, categories, stores, store groups, and customers are eligible for the promotion. The template rules decide which customer activities are eligible for the promotion and the rewards available for each eligible customer activity.	Small, 61.0	61.0
description	String	The description of the promotion rule.	Small, 61.0	61.0
apiName	String	The API name of the promotion rule.	Small, 61.0	61.0
isEligibilityRule	Boolean	Indicates whether the rule is the promotion eligibility rule (true) or not (false).	Small, 61.0	61.0
sequenceNumber	Integer	The sequence number of the rule in the promotion rules list.	Small, 61.0	61.0
<del>ruleCriteriaList</del> ruleFilterCriteriaList	Rule Filter Criteria List []	The list of filter criteria that belong to the rule.	Small, 61.0	61.0
ruleActionList	Rule Action List []	The list of actions that belong to the rule.	Small, 61.0	61.0
ruleParameterList	Rule Parameter List []	The list of parameters that belong to the rule.	Small, 61.0	61.0

**Table 6: Rule Filter Criteria Output**

Property Name	Type	Description	Filter Group and Version	Available Version
name	String	The name of the promotion rule's filter criteria.	Small, 61.0	61.0
type	String	The type of promotion rule filter criteria.	Small, 61.0	61.0
conditionLogic	String	The condition logic of the promotion rule's filter criteria.	Small, 61.0	61.0
sequenceNumber	Integer	The sequence number of the promotion rule's filter criteria in the promotion rule.	Small, 61.0	61.0
parentFilterCriteria	String	The parent filter criteria that the promotion rule filter criteria belongs to.	Small, 61.0	61.0
ruleConditionList	Rule Condition List []	The list of conditions that belong to the promotion rule filter criteria.	Small, 61.0	61.0
ruleActionList	Rule Action List []	The list of actions that belong to the promotion rule filter criteria.	Small, 61.0	61.0

**Table 7: Rule Condition Output**

Property Name	Type	Description	Filter Group and Version	Available Version
parameterName	String	The name of the parameter that's used to define the rule filter criteria condition.	Small, 61.0	61.0
operator	String	The operator of the rule filter criteria condition.	Small, 61.0	61.0
valueList	Map<String,String>	The value that the rule filter criteria checks the parameter against.	Small, 61.0	61.0
valueType	String	The type of value.	Small, 61.0	61.0
sequenceNumber	Integer	The sequence number of the rule filter criteria condition in the rule condition list.	Small, 61.0	61.0
matchType	String	Specifies whether the condition qualifies as true when all the values in the value list match the condition's requirements or when any one of the values in the value list match the condition's requirements.	Small, 61.0	61.0

**Table 8: Rule Action Output**

Property Name	Type	Description	Filter Group and Version	Available Version
name	String	The name of the action associated with either a promotion rule or a promotion rule criteria.	Small, 61.0	61.0
sequenceNumber	Integer	The sequence number of the action in either a promotion rule or a promotion rule criteria.	Small, 61.0	61.0
actionType	String	The type of promotion rule action.	Small, 61.0	61.0
ruleActionParameterList	Rule Action Parameter List []	The list of action parameters that belong to the rule or criteria action.	Small, 61.0	61.0

**Table 9: Rule Action Parameter Output**

Property Name	Type	Description	Filter Group and Version	Available Version
name	String	The name of the parameter associated with the promotion rule action.	Small, 61.0	61.0
value	String	The value of the promotion rule action parameter.	Small, 61.0	61.0
sequenceNumber	Integer	The sequence number of the promotion rule action parameter.	Small, 61.0	61.0
valueType	String	The type of value.	Small, 61.0	61.0
relatedCriteriaName	String	The name of the rule filter criteria associated with the rule action parameter.	Small, 61.0	61.0

**Table 10: Rule Parameter Output**

Property Name	Type	Description	Filter Group and Version	Available Version
name	String	The name of the parameter associated with the rule.	Small, 61.0	61.0
dataType	String	The data type of the rule parameter.	Small, 61.0	61.0
type	String	The type of rule parameter.	Small, 61.0	61.0
decimalPlaces	Integer	The number of decimal places supported by a numeric type rule parameter.	Small, 61.0	61.0
value	String	The value of the rule parameter.	Small, 61.0	61.0
isInput	Boolean	Indicates whether the parameter stores a value that's used to process the rule (true) or not (false).	Small, 61.0	61.0

Property Name	Type	Description	Filter Group and Version	Available Version
isOutput	Boolean	Indicates whether the parameter stores a value that's provided after the rule is processed (true) or not (false).	Small, 61.0	61.0
isCollection	Boolean	Indicates whether the parameter stores a list of comma-separated values (true) or not (false).	Small, 61.0	61.0
contextTag	String	The context tag that is used in the rule parameter.The context tag belongs to the context definition associated with the promotion's rule library.	Small, 61.0	61.0
hasContextTagMapping	Boolean	Indicates whether the context tag has a corresponding context tag mapping (true) or not (false).The context mapping is available in the active version of the rule library associated with the promotion.	Small, 61.0	61.0
aggregateFunction	String	The aggregate function that's used in the rule parameter.	Small, 61.0	61.0
groupByContextTag	String	The customer activity inputs are grouped by the context tag.	Small, 61.0	61.0
relatedFilterCriteriaName	String	The name of the rule filter criteria associated with the rule parameter.	Small, 61.0	61.0

## Clone Promotion Output

Output representation of Clone Promotion API.

### JSON Example

```
{
  "promotionId": "01pxx00000000EXAAQ"
}
```

Property Name	Type	Description	Available Version
promotionId	String	The ID of the new promotion.	61.0

## Promotion Coupon Availability

Output representation of the promotion coupon availability.



Property Name	Type	Description	Filter Group and Version	Available Version
coupon	<a href="#">Promotion Coupon</a> on page 441	The details of the coupon that is eligible for the promotion.	Small, 60.0	60.0
couponAvailabilityMessage	<a href="#">Coupon Availability Message</a> on page 441	Specifies reasons for coupon ineligibility for the promotion.	Small, 60.0	60.0

## Promotion Coupon

Output representation of the promotion coupon.

Property Name	Type	Description	Filter Group and Version	Available Version
couponCode	String	The unique code of the coupon.	Small, 60.0	60.0
startDateTime	DateTime	The start date and time of the coupon.	Small, 60.0	60.0
endDateTime	DateTime	The end date and time of the coupon.	Small, 60.0	60.0
status	String	The status of the coupon.	Small, 60.0	60.0

## Coupon Availability Message

Output representation of the coupon availability message.

Property Name	Type	Description	Filter Group and Version	Available Version
MultipleActiveCoupons	String	There are multiple active coupons that are eligible for the cart on the activity date.	Small, 62.0	62.0
NoActiveCoupon	String	No coupons are active on the cart activity date	Small, 62.0	62.0
SingleActiveLimitReached	String	Only one coupon is active on the cart activity date but it has reached its usage limit.	Small, 62.0	62.0

## Coupon Usage Result

Represents the response for coupon usage increase and decrease POST method.

### Sample Response

```
[
  {
    "couponCode": "QWERTY",
    "isSuccess": true,
  }
]
```

```

    "redemptionLimit": 10,
    "redemptionLimitPerCustomer": 3,
    "remainingRedemptionCount": 6,
    "remainingRedemptionCountForCustomer": 1
  },
  {
    "couponCode": "ABCD",
    "errorMessage": "Coupon code doesn't exist.",
    "isSuccess": false
  }
]


```

Property Name	Type	Description	Filter Group and Version	Available Version
couponCode	String	The code of the coupon for which usage has been increased or decreased.	Small, 62.0	62.0
redemptionLimit	Integer	The total number of times customers can redeem the coupon.	Small, 62.0	62.0
remainingRedemptionCount	Integer	The remaining number of times the coupon can be redeemed.	Small, 62.0	62.0
redemptionLimitPerCustomer	Integer	The total number of times each customer can redeem the coupon.	Small, 62.0	62.0
remainingRedemptionCountForCustomer	Integer	The remaining number of times the specified customer can redeem the coupon.	Small, 62.0	62.0
isSuccess	Boolean	Indicates whether the usage of the coupon was successfully increased or decreased.	Small, 62.0	62.0
errorMessage	String	The error message that contains the details of the why the coupon usage wasn't updated.	Small, 62.0	62.0

## Member Engagement Trail Output

Output representation of member engagement trail API.

### Sample Response

 **Note:** The field `remainingResetCount` and `totalResetCount` may show null if the set limit is 1.

```

{
  "memberEngagementAttributeOutputRepresentations": [
    {
      "attributeStatus": "Active",
      "currentValue": null,
      "defaultValue": "0",
      "description": null,
      "endDate": "2025-01-31",
      "name": "Book A Cab",

```

```

    "startDate": "2024-11-01",
    "targetValue": "2.0"
  },
  {
    "attributeStatus": "Active",
    "currentValue": null,
    "defaultValue": "0",
    "description": null,
    "endDate": "2025-01-31",
    "name": "Book A Hotel Room",
    "startDate": "2024-11-01",
    "targetValue": "1.0"
  },
  {
    "attributeStatus": "Active",
    "currentValue": null,
    "defaultValue": "0",
    "description": null,
    "endDate": "2025-01-31",
    "name": "Book A Flight",
    "startDate": "2024-11-01",
    "targetValue": "1.0"
  }
],
"message": null,
"promotionName": "Engagement Trail",
"remainingResetCount": null,
"rewards": [
  {
    "badgeReward": [],
    "pointsRewards": [
      {
        "currencyName": "Reward Points",
        "points": 200
      },
      {
        "currencyName": "Tier Points",
        "points": 100
      }
    ]
  },
  {
    "voucherRewards": []
  }
],
"status": true,
"totalResetCount": null
}

```

Property Name	Type	Description	Filter Group and Version	Available Version
status	Boolean	Indicates whether the request succeeded (true) or failed (false).	Small, 62.0	62.0

Property Name	Type	Description	Filter Group and Version	Available Version
message	String	The message explaining why the request failed.	Small, 62.0	62.0
engagementAttribute		Indicates the event that is tracked for a member.	Small, 62.0	62.0
totalResetCount	Integer	The maximum number of times a member can complete the engagement trail and earn rewards.	Small, 62.0	62.0
remainingResetCount	Integer	The number of times a member can still complete the engagement trail and earn rewards.	Small, 62.0	62.0

## Member Engagement Attribute

Output representation of Member Engagement Attribute.

Property Name	Type	Description	Filter Group and Version	Available Version
name	String	The name of the engagement attribute.	Small, 62.0	62.0
description	String	The description of the engagement attribute.	Small, 62.0	62.0
startDate	DateTime	The date from when the engagement attribute is tracked for members.	Small, 62.0	62.0
endDate	DateTime	The date until which the engagement attribute is tracked for members.	Small, 62.0	62.0
currentValue	Integer	The current value of the engagement attribute.	Small, 62.0	62.0
targetValue	Integer	The value that members have to achieve to complete the engagement attribute's target.	Small, 62.0	62.0
attributeStatus	String	The status of the engagement attribute.	Small, 62.0	62.0
defaultValue	Integer	The default value of the engagement attribute.	Small, 62.0	62.0
pointsRewards	List	Indicates the rewards in the form of points or currencies.	Small, 62.0	62.0
currencyId	String	The ID of the currency.	Small, 62.0	62.0
currencyName	String	The name of the currency.	Small, 62.0	62.0
points	Integer	Number of points to give as a reward.	Small, 62.0	62.0

Property Name	Type	Description	Filter Group and Version	Available Version
voucherRewards	List	Indicates the rewards in the form of vouchers.	Small, 62.0	62.0
voucherDefinitionId	String	The ID of the voucher definition.	Small, 62.0	62.0
voucherName	String	The name of the voucher.	Small, 62.0	62.0
BadgeRewards	List	Indicates the rewards in the form of badges.	Small, 62.0	62.0
badgeId	String	The ID of the badge.	Small, 62.0	62.0
badgeName	String	The name of the badge.	Small, 62.0	62.0

## Loyalty Management Connect API Interchangeable Request Fields

The Loyalty Management Business API supports interchangeable fields in JSON requests. For some of the request fields, you can use either an ID, number, or name. For example, if you don't want to store and use Salesforce IDs when accessing Loyalty Management-related data from an external system, simply use names and numbers instead.

Here's a list of Loyalty Management object fields that are interchangeable with alternative field names.

Field Name	Alternative Field Name
MemberId	MembershipNumber
VoucherId	VoucherCode or VoucherNumber Ensure that the voucher code is unique among all active vouchers.
LoyaltyProgramId	LoyaltyProgramName
JournalTypeId	JournalTypeName
JournalSubTypeId	JournalSubTypeName
PromotionId	PromotionName Ensure that the promotion name is unique among all active promotions.
ProductId	ProductCode or ProductName Ensure that the product name or code is unique among all active products.
ProductCategoryId	ProductCategoryName Ensure that the product name or code is unique among all active products.
PartnerId	PartnerName

**JSON example with IDs**

```
{
  "MemberId": "01MT10000004CfA",
  "JournalTypeId": "01ET10000004CQp",
  "JournalSubTypeId": "0pQS10000005CQm",
  "VoucherId": "0kdxx000000000B245",
  "ProductId": "01Vxx000000000cj",
  "ProductCategoryId": "0c8RM0000004DnXYAU",
  "PromotionId": "0c8xx000000000BJEAY",
  "LoyaltyProgramId": "01pT10000004CdN"
}
```

**JSON example with names and numbers**

```
{
  "MembershipNumber": "Member123",
  "JournalType": "Accrual",
  "JournalSubType": "Social",
  "VoucherCode": "15OFF",
  "ProductName": "Nike",
  "ProductCategoryName": "Apparel",
  "PromotionName": "Flat20",
  "LoyaltyProgramName": "NTO"
}
```

## CHAPTER 7 Loyalty Management Standard Invocable Actions

### In this chapter ...

- [Adjust Points](#)
- [Assign Member Tier Benefits](#)
- [Accrual Cancellation](#)
- [Credit Points](#)
- [Change Tier](#)
- [Change All Tiers or None](#)
- [Debit Points](#)
- [Generate Referral Code](#)
- [Get Member's Active Segments](#)
- [Get Points Balance](#)
- [Get Promotions Based on Member's Market Segment](#)
- [Get Promotions Based on Transaction Journal](#)
- [Get Tier](#)
- [Issue Voucher](#)
- [Merge Memberships Action](#)
- [Process Member Benefit](#)
- [Redemption Cancellation](#)
- [Run Program Process for Transaction Journal](#)
- [Transfer Member Points to Groups](#)
- [Unmerge Memberships Action](#)

Create and manage loyalty programs for your organization using invocable actions. Use tier groups to group members and change member tier based on their loyalty points, assign and process benefits based on their tier, manage loyalty points for a member, issue voucher for a member, and cancel redemption transactions for a member for Loyalty Management. For more information on standard invocable actions, see [REST API Developer Guide](#) and [Actions Developer Guide](#).

## Loyalty Management Standard Invocable Actions

- Update Member Progress for Cumulative Promotion



## Adjust Points

---

Adjust loyalty points for a specified program member or journal transaction.

This action is available in API version 51.0 and later for users with Loyalty Management license.

### Supported REST HTTP Methods

#### URI

`/services/data/vXX.X/actions/standard/adjustPointsAction`

#### Formats

JSON, XML

#### HTTP Methods

POST

#### Authentication

Authorization: Bearer *token*

### Inputs

Input	Details
AdjustmentCategory	<p><b>Type</b> String</p> <p><b>Description</b> The category of adjustment.</p>
AdjustmentType	<p><b>Type</b> string</p> <p><b>Description</b> Required. The type of adjustment required.</p>
JournalId	<p><b>Type</b> ID</p> <p><b>Description</b> ID of the transaction journal.</p>
LoyaltyProgramCurrencyId	<p><b>Type</b> ID</p> <p><b>Description</b> ID of the loyalty program currency.</p>
LoyaltyProgramMemberId	<p><b>Type</b> ID</p>

Input	Details
	<p><b>Description</b> Required. ID of the loyalty program member to whom the points must be adjusted.</p>
Points	<p><b>Type</b> Float</p> <p><b>Description</b> Required. The points available for the loyalty program member.</p>
PointsExpirationDate	<p><b>Type</b> string</p> <p><b>Description</b> The date on which fixed-model non-qualifying points credited by the action expire. Points expiration date is only applicable when the adjustment is crediting points to the member. This input is available in API version 55.0 and later.</p>
ReasonforChange	<p><b>Type</b> string</p> <p><b>Description</b> Required. The reason for adjustment.</p>

## Outputs

Output	Details
AdjustmentTransactionJournalId	<p><b>Type</b> string</p> <p><b>Description</b> The ID of the adjustment transaction journal generated after the request is successful.</p>

## Example

### Sample Request

```
{
  "inputs": [{
    "LoyaltyProgramMemberId": "01mxx00004000001AAB",
    "LoyaltyProgramCurrencyId": "01pxx00004000001AAB",
    "Points": 500.0,
    "PointsExpirationDate": "2022-08-12",
    "AdjustmentType": "Credit",
    "ReasonForChange": "Compensation"
  }]
}
```

**Sample Response**

```
{
  "errors":null,
  "isSuccess":true,
  "outputValues":{
    "AdjustmentTransactionJournalId":"01vxx00004000001AAB"}
}
```

## Assign Member Tier Benefits

---

Assign benefits to a loyalty program members based on their member's tier for benefits that are associated with a Benefit Action.

This action is available in API version 51.0 and later for users with Loyalty Management license.

## Supported REST HTTP Methods

**URI**

/services/data/v**XX.X**/actions/standard/assignMemberTierBenefits

**Formats**

JSON, XML

**HTTP Methods**

POST

**Authentication**

Authorization: Bearer *token*

## Inputs

Input	Details
LoyaltyProgramMemberId	<p><b>Type</b> ID</p> <p><b>Description</b> Required. ID of the loyalty program member to associate with the benefits.</p>

## Outputs

Output	Details
MemberBenefits	<p><b>Type</b> string[]</p> <p><b>Description</b> The benefits associated with the member.</p>

## Example

### Sample Request

```
{
  "inputs": [{
    "LoyaltyProgramMemberId": "01MR0000000001OAA"
  }]
}
```

### Sample Response

```
{
  "actionName": "assignTierBenefits",
  "errors": null,
  "isSuccess": true,
  "outputValues": {
    "MemberBenefits": [
      '0hWR0000000001OAA',
      '0hWR0000000001OAB',
      '0hWR0000000001OAC'
    ]
  }
}
```

## Accrual Cancellation

---

Cancel a specific set of accrual transactions.

This action is available in API version 51.0 and later for users with Loyalty Management license.

## Supported REST HTTP Methods

### URI

/services/data/v**xx.x**/actions/standard/cancelAccrual

### Formats

JSON, XML

### HTTP Methods

POST

### Authentication

Authorization: Bearer *token*

## Inputs

Input	Details
CancellationReason	Type string

---

Input	Details
	<p><b>Description</b> Required. The reason for accrual cancellation.</p>
ReversalCategory	<p><b>Type</b> string</p> <p><b>Description</b> The category of accrual cancellation.</p>
TransactionJournalId	<p><b>Type</b> string</p> <p><b>Description</b> Required. ID of the transaction journal.</p>

## Outputs

Output	Details
CancellationTransactionJournalId	<p><b>Type</b> string</p> <p><b>Description</b> The ID of the cancellation transaction journal generated after the request is successful.</p>

## Example

### Sample Request

```
{
  "inputs": [{
    "TransactionJournalId": "01vxx00004000001AAA",
    "CancellationReason": "Duplicate Accrual"
  }]
}
```

### Sample Response

```
{
  "errors": null,
  "isSuccess": true,
  "outputValues": {
    "CancellationTransactionJournalId": "01vxx00004000001AAB"
  }
}
```

## Credit Points

---

Credit loyalty points to a specified loyalty program member's balance.

This action is available in API version 51.0 and later for users with Loyalty Management license.

### Supported REST HTTP Methods

#### URI

`/services/data/vXX.X/actions/standard/creditPoints`

#### Formats

JSON, XML

#### HTTP Methods

POST

#### Authentication

Authorization: Bearer *token*

### Inputs

Input	Details
AdditionalNotes	<p><b>Type</b> String</p> <p><b>Description</b> Additional information about the transactions performed by the loyalty program members.</p>
CorrelationId	<p><b>Type</b> ID</p> <p><b>Description</b> ID of any Salesforce entity.</p>
FlowId	<p><b>Type</b> ID</p> <p><b>Description</b> ID of the flow associated with credit points action.</p>
JournalId	<p><b>Type</b> ID</p> <p><b>Description</b> Required. ID of the transaction journal.</p>
LoyaltyProgramMemberId	<p><b>Type</b> ID</p>

Input	Details
	<p><b>Description</b> Required. ID of the loyalty program member.</p>
Points	<p><b>Type</b> Float</p> <p><b>Description</b> Required. The points available for the loyalty program member.</p>
ProgramCurrencyName	<p><b>Type</b> string</p> <p><b>Description</b> Required. The name of the program currency.</p>
PointsExpirationDate	<p><b>Type</b> string</p> <p><b>Description</b> The date on which fixed-model non-qualifying points credited by the action expire. This input is available in API version 55.0 and later.</p>
ProgramName	<p><b>Type</b> string</p> <p><b>Description</b> Required. The name of the loyalty program.</p>
LoyaltyProgramCurrencySubtypeName	<p><b>Type</b> string</p> <p><b>Description</b> Required. The name of the loyalty program currency subtype that is used to credit points from the member.</p>

## Outputs

Output	Details
JournalId	<p><b>Type</b> ID</p> <p><b>Description</b> ID of the transaction journal.</p>
LoyaltyProgramMemberId	<p><b>Type</b> string</p>

Output	Details
	<p><b>Description</b> ID of the loyalty program member.</p>
ProgramCurrencyName	<p><b>Type</b> string</p> <p><b>Description</b> The name of the program currency.</p>
ProgramName	<p><b>Type</b> string</p> <p><b>Description</b> The name of the loyalty program.</p>

## Example

### Sample Request

```
{
  "inputs": [{
    "LoyaltyProgramMemberId": "01Mxx0000A000001EAA",
    "ProgramCurrencyName": "Avios",
    "ProgramName": "baLoyaltyProgram",
    "Points": 500.0,
    "PointsExpirationDate": "12-08-2022",
    "CorrelationId": "01cxx00004000002AAA",
    "JournalId": "01cxx00204000001AAA",
    "FlowId": "01cxx00204000001AAA",
    "AdditionalNotes": "Any additional Information"
  }]
}
```

### Sample Response

```
{
  "errors": null,
  "isSuccess": true,
  "outputValues": {
    "LoyaltyProgramMemberId": "01Mxx0000A000001EAA",
    "ProgramCurrencyName": "Avios",
    "ProgramName": "baLoyalty",
    "JournalId": "01Mxx0000A000001EAA"
  }
}
```

## Change Tier

Change the tier for a specified loyalty program member.



This action is available in API version 51.0 and later for users with Loyalty Management license.

## Supported REST HTTP Methods

### URI

/services/data/v**XX.X**/actions/standard/changeTier

### Formats

JSON, XML

### HTTP Methods

POST

### Authentication

Authorization: Bearer *token*

## Inputs

Input	Details
ChangeReason	<p><b>Type</b> string</p> <p><b>Description</b> The reason for change in member tier.</p>
LoyaltyProgramMemberId	<p><b>Type</b> ID</p> <p><b>Description</b> Required. ID of the loyalty program member.</p>
ProgramName	<p><b>Type</b> string</p> <p><b>Description</b> Required. The loyalty program name associated with the tier.</p>
TierGroupName	<p><b>Type</b> string</p> <p><b>Description</b> Required. The name of the tier group.</p>
TargetTierName	<p><b>Type</b> string</p> <p><b>Description</b> Required. The new tier to be associated with the loyalty program member</p>

## Outputs

Output	Details
LoyaltyProgramMemberId	<p><b>Type</b> string</p> <p><b>Description</b> ID of the loyalty program member.</p>
ProgramName	<p><b>Type</b> string</p> <p><b>Description</b> The name of the loyalty program.</p>
TargetTierName	<p><b>Type</b> string</p> <p><b>Description</b> The new tier associated with the loyalty program member.</p>
TierGroupName	<p><b>Type</b> string</p> <p><b>Description</b> The name of the tier group.</p>

## Example

### Sample Request

```
{
  "inputs": [{
    "TierGroupName": "QualifyingGroup",
    "ProgramName": "baProgram",
    "TargetTierName": "Gold",
    "LoyaltyProgramMemberId": "01cxx00006000001AAA",
    "ChangeReason": "Reason to change the tier"
  }]
}
```

### Sample Response

```
{
  "errors": null,
  "isSuccess": true,
  "outputValues": {
    "LoyaltyprogramMemberId": ".01Mxx0000A000001EAA",
    "TierGroupName": "QualifyingGroup",
    "ProgramName": "baProgram",
    "TargetTierName": "Gold"
  }
}
```

```
}
}
```

## Change All Tiers or None

---

Change tier for a specified loyalty program member only when all the input parameters meet the criteria.

This action is available in API version 51.0 and later for users with Loyalty Management license.

### Supported REST HTTP Methods

#### URI

`/services/data/vxx.x/actions/standard/changeAllTierOrNone`

#### Formats

JSON, XML

#### HTTP Methods

POST

#### Authentication

Authorization: Bearer *token*

### Inputs

Input	Details
ChangeReason	<p><b>Type</b> string</p> <p><b>Description</b> The reason for change in the tier.</p>
ChangeReasonType	<p><b>Type</b> string</p> <p><b>Description</b> The reason type for change in the tier.</p>
LoyaltyProgramMemberId	<p><b>Type</b> ID</p> <p><b>Description</b> Required. ID of the loyalty program member.</p>
ProgramName	<p><b>Type</b> string</p> <p><b>Description</b> Required. The name of the loyalty program.</p>

Input	Details
TierGroupName	<p><b>Type</b> string</p> <p><b>Description</b> Required. The name of the tier group.</p>
TargetTierName	<p><b>Type</b> string</p> <p><b>Description</b> Required. The new tier to be associated with the loyalty program member.</p>

## Outputs

Output	Details
LoyaltyProgramMemberId	<p><b>Type</b> ID</p> <p><b>Description</b> ID of the loyalty program member.</p>
ProgramName	<p><b>Type</b> string</p> <p><b>Description</b> The name of the loyalty program.</p>
TierGroupName	<p><b>Type</b> string</p> <p><b>Description</b> The name of the tier group.</p>
TargetTierName	<p><b>Type</b> string</p> <p><b>Description</b> The new tier associated with the loyalty program.</p>

## Example

### Sample Request

```
{
  "inputs": [{
    "TierGroupName": "Default Tier Group",
```

```

    "ProgramName":"Rewards program",
    "TargetTierName":"Gold",
    "LoyaltyProgramMemberId": "01cxx00006000001AAA",
    "ChangeReason":"Reason to change the tier"
  }}
}

```

### Sample Response

```

{
  "actionName":"changeTierWhenNoErrors",
  "errors":null,
  "isSuccess":true,
  "outputValues":{
    "LoyaltyProgramMemberId":"01cxx00006000001AAA",
    "TierGroupName":"Default Tier Group",
    "ProgramName": "Rewards program",
    "TargetTierName":"Gold",
  }
}

```

## Debit Points

Debit loyalty points from a specified loyalty program member's balance. You can debit loyalty points to redeem for a service or upgrade the member to a higher tier.



**Note:** This action updates the point balance in the LoyaltyProgramMemberCurrency table in your Salesforce org.

This action is available in API version 51.0 and later for users with Loyalty Management license.

## Supported REST HTTP Methods

### URI

/services/data/v**XX.X**/actions/standard/debitPoints

### Formats

JSON, XML

### HTTP Methods

POST

### Authentication

Authorization: Bearer *token*

## Inputs

Input	Details
AdditionalNotes	Type String

Input	Details
	<p><b>Description</b> Additional information about the transactions performed by the loyalty program members.</p>
CorrelationId	<p><b>Type</b> ID</p> <p><b>Description</b> ID of any Salesforce entity.</p>
FlowId	<p><b>Type</b> ID</p> <p><b>Description</b> ID of the flow associated with the debit points action.</p>
JournalId	<p><b>Type</b> ID</p> <p><b>Description</b> Required. ID of the transaction journal.</p>
LoyaltyProgramMemberId	<p><b>Type</b> ID</p> <p><b>Description</b> Required. ID of the loyalty program member to whom the points must be deducted.</p>
Points	<p><b>Type</b> Float</p> <p><b>Description</b> Required. The points available for the loyalty program member.</p>
ProgramCurrencyName	<p><b>Type</b> string</p> <p><b>Description</b> Required. The name of the loyalty program currency.</p>
ProgramName	<p><b>Type</b> string</p> <p><b>Description</b> Required. The name of the loyalty program.</p>
LoyaltyProgramCurrencySubtypeName	<p><b>Type</b> string</p>

**Input****Details****Description**

Required. The name of the loyalty program currency subtype that is used to debit points from the member.

## Outputs

**Output****Details**

JournalId

**Type**

ID

**Description**

ID of the transaction journal.

LoyaltyProgramMemberId

**Type**

string

**Description**

ID of the loyalty program member.

PointsBalance

**Type**

Integer

**Description**

The total points available for the loyalty program member.

ProgramCurrencyName

**Type**

string

**Description**

The name of the program currency.

ProgramName

**Type**

string

**Description**

The name of the loyalty program.

## Example

**Sample Request**

```
{
  "inputs": [{
    "LoyaltyProgramMemberId": "01Mxx0000A000001EAA",
    "ProgramCurrencyName": "Avios",
    "ProgramName": "baLoyaltyProgram",
  }
]
```

```

    "Points": 500.0,
    "CorrelationId": "01cxx00004000002AAA",
    "JournalId": "01cxx00204000001AAA",
    "AdditionalNotes": "Any additional Information"
  }}
}

```

### Sample Response

```

[ {
  "actionName" : "debitPoints",
  "errors" : null,
  "isSuccess" : true,
  "outputValues" : {
    "JournalId" : "01cxx00204000001AAA",
    "PointsBalance" : 130.0,
    "ProgramCurrencyName" : "exclusive points",
    "LoyaltyProgramMemberId" : "01Mxx0000A000001EAA",
    "ProgramName" : "sample program"
  }
} ]

```

## Generate Referral Code

---

Generate a referral code for a loyalty program member.



**Note:** If a loyalty program member already has a referral code, generating another code replaces the existing one.

This action is available in API version 57.0 and later for users with the Loyalty Management license.

## Supported REST HTTP Methods

### URI

/services/data/v**XX.X**/actions/standard/generateMemberReferralCode

### Formats

JSON

### HTTP Methods

POST

### Authentication

Authorization: Bearer *token*

## Inputs

Input	Details
LoyaltyProgramMemberId	Type String



**Input****Details****Description**

Required. ID of the loyalty program member to generate the referral code for.



**Note:** Specify an array of member IDs in the JSON request to generate referral codes for multiple members.

## Outputs

None.

## Example

**Sample Request**

```
{
  "inputs": [{
    "LoyaltyProgramMemberId": "01Mxx0000A000001EAA"
  }, {
    "LoyaltyProgramMemberId": "01Qxx0000B000001EBS"
  }]
}
```

## Get Member's Active Segments

Retrieve active Data Cloud market segments that a loyalty program member is a part of.

For more information about how the `getMemberActiveSegments` action can be used to process transaction journals and apply promotions, see [Get a List of Member's Active Segments for a Member](#) in Salesforce Help.

This action is available in API version 55.0 and later for users in orgs where either B2C - Loyalty, B2C - Loyalty Plus, Loyalty Management - Growth, or Loyalty Management - Plus license and Data Cloud are enabled.

## Supported REST HTTP Methods

**URI**

`/services/data/vXX.X/actions/standard/getMemberActiveSegments`

**Formats**

JSON, XML

**HTTP Methods**

POST

**Authentication**

Authorization: Bearer *token*

## Inputs

Input	Details
loyaltyProgramMemberID	<p><b>Type</b> ID</p> <p><b>Description</b> Required. The ID of the loyalty program member to retrieve associated active segments for.</p>
segmentIds	<p><b>Type</b> ID</p> <p><b>Description</b> A comma-delimited list of segmentIds to retrieve for the specified member's associated segments.</p>

## Outputs

Output	Details
memberSegmentIDs	<p><b>Type</b> string</p> <p><b>Description</b> A comma-delimited list of the active segment IDs that the specified loyalty program member is a part of.</p>

## Example

### Sample Request

```
{
  "inputs": [ {
    "loyaltyProgramMemberID": "01Vxx000000016GMAQ"
  } ]
}
```

### Sample Response

```
{
  "errors": null,
  "isSuccess": true,
  "outputValues": {
    "memberSegmentIDs": "01vxx00004000001AAB"
  }
}
```

# Get Points Balance

---

Retrieve the loyalty points balance for a specified loyalty program member.

This action is available in API version 51.0 and later for users with Loyalty Management license.

## Supported REST HTTP Methods

### URI

/services/data/v**XX.X**/actions/standard/getPointsBalance

### Formats

JSON, XML

### HTTP Methods

POST

### Authentication

Authorization: Bearer *token*

## Inputs

Input	Details
LoyaltyProgramMemberId	<b>Type</b> string <b>Description</b> Required. ID of the loyalty program member to whom the points balance are to be fetched.
ProgramCurrencyName	<b>Type</b> string <b>Description</b> Required. The name of the program currency.
ProgramName	<b>Type</b> string <b>Description</b> Required. The name of the loyalty program.

## Outputs

Output	Details
LoyaltyProgramMemberId	<b>Type</b> ID

Output	Details
	<p><b>Description</b> ID of the loyalty program member.</p>
PointsBalance	<p><b>Type</b> integer</p> <p><b>Description</b> The points available for the loyalty program member.</p>
ProgramCurrencyName	<p><b>Type</b> string</p> <p><b>Description</b> The name of the program currency.</p>
ProgramName	<p><b>Type</b> string</p> <p><b>Description</b> The name of the loyalty program.</p>
EscrowPointsBalance	<p><b>Type</b> double</p> <p><b>Description</b> The number points that are placed in escrow for the member. This field is available in API version 53.0 and later.</p>

## Example

### Sample Request

```
{
  "inputs": [{
    "LoyaltyProgramMemberId": "01Mxx0000A000001EAA",
    "ProgramCurrencyName": "Avios",
    "ProgramName": "baLoyalty"
  }]
}
```

### Sample Response

```
{
  "errors": null,
  "isSuccess": true,
  "outputValues": {
    "LoyaltyprogramMemberId": "01Mxx0000A000001EAA",
    "ProgramCurrencyName": "Avios",
    "ProgramName": "baLoyalty"
    "PointsBalance": 500.0
  }
}
```

```

    "EscrowPointsBalance": 25.00
  }
}

```

## Get Promotions Based on Member's Market Segment

---

Retrieve which promotions are eligible for a member based on the member's market segment.

For more information about how the `getLoyaltyPromotionBasedOnSalesforceCDP` action filters eligible promotions, see [Get a List of Promotions Applicable for a Member](#) in Salesforce Help.

This action is available in API version 53.0 and later for users in orgs where either B2C - Loyalty, B2C - Loyalty Plus, Loyalty Management - Growth, or Loyalty Management - Advanced license and Data Cloud are enabled.

### Supported REST HTTP Methods

#### URI

`/services/data/v $xx.x$ /actions/standard/getLoyaltyPromotionBasedOnSalesforceCDP`

#### Formats

JSON, XML

#### HTTP Methods

POST

#### Authentication

Authorization: Bearer *token*

### Inputs

Input	Details
MemberId	<p><b>Type</b> ID</p> <p><b>Description</b> Required. ID of the loyalty program member for whom promotions must be filtered. Ensure that the member is part of a Data Cloud market segment.</p>
PromotionRecords	<p><b>Type</b> sObject</p> <p><b>Description</b> Promotion records to filter the promotions that are applicable to the member's market segment.</p>

## Outputs

Output	Details
PromotionRecords	<p><b>Type</b> sObject</p> <p><b>Description</b> Promotion records that the member is eligible for based on the member's market segment.</p>

## Example

### Sample Request

```
{
  "inputs": [ {
    "MemberId": "01Mxx0000000EYAMA2"
  } ]
}
```

### Sample Response

```
[ {
  "actionName" : "getLoyaltyPromotionBasedOnSalesforceCDF",
  "errors" : null,
  "isSuccess" : true,
  "outputValues" : {
    "PromotionRecords" : [ {
      "attributes" : {
        "type" : "Promotion",
        "url" : "/services/data/v53.0/subjects/Promotion/0c8T10000004CBEIA2"
      },
      "Id" : "0c8xx0000004CBEIA2",
      "OwnerId" : "005xx000000HwDoIAK",
      "IsDeleted" : false,
      "Name" : "Sample Promotion",
      "CreatedDate" : "2021-07-21T04:13:15.000+0000",
      "CreatedById" : "005xx000000HwDoIAK",
      "LastModifiedDate" : "2021-07-21T04:13:15.000+0000",
      "LastModifiedById" : "005xx000000HwDoIAK",
      "SystemModstamp" : "2021-07-21T04:13:15.000+0000",
      "LastViewedDate" : "2021-07-21T04:15:21.000+0000",
      "LastReferencedDate" : "2021-07-21T04:15:21.000+0000",
      "StartDate" : "2021-07-20",
      "IsActive" : true,
      "LoyaltyPromotionType" : "Cumulative",
      "IsEnrollmentRequired" : false,
      "CumulativeUsageTarget" : 100,
      "FulfillmentAction" : "CreditPoints",
      "LoyaltyProgramId" : "01pxx00000000HWIAY",
      "TotalRewardPoints" : 100,
      "LoyaltyProgramCurrencyId" : "01cxx00000000XeIAI",
      "TransactionJournalType" : "Accrual",
    } ]
  }
}
```

```

        "IsPointsWithFactor" : false
      } ]
    }
  } ]

```

## Get Promotions Based on Transaction Journal

---

Retrieve active promotions that are eligible for a transaction journal. The action can process accrual or redemption transactions.

For more information about how the `getLoyaltyPromotion` action filters eligible promotions, see [Get a List of Promotions Applicable for a Transaction](#) in Salesforce Help.

This action is available in API version 53.0 and later for users in orgs where either B2C - Loyalty, B2C - Loyalty Plus, Loyalty Management - Growth, or Loyalty Management - Advanced license is enabled.

### Supported REST HTTP Methods

#### URI

`/services/data/v $xx.x$ /actions/standard/getLoyaltyPromotion`

#### Formats

JSON, XML

#### HTTP Methods

POST

#### Authentication

Authorization: Bearer *token*

### Inputs

Input	Details
JournalId	<p><b>Type</b> ID</p> <p><b>Description</b> Required. ID of an Accrual or Redemption type Transaction Journal record.</p>

### Outputs

Output	Details
PromotionRecords	<p><b>Type</b> sObject</p> <p><b>Description</b> Promotion records that are eligible for the transaction journal.</p>

## Example

### Sample Request

```
{
  "inputs": [ {
    "JournalId": "01Vxx000000016GMAQ"
  } ]
}
```

### Sample Response

```
[ {
  "actionName" : "getLoyaltyPromotion",
  "errors" : null,
  "isSuccess" : true,
  "outputValues" : {
    "PromotionRecords" : [ {
      "attributes" : {
        "type" : "Promotion",
        "url" : "/services/data/v53.0/subjects/Promotion/0c8T10000004CBEIA2"
      },
      "Id" : "0c8xx0000004CBEIA2",
      "OwnerId" : "005xx000000HwDoIAK",
      "IsDeleted" : false,
      "Name" : "Sample Promotion",
      "CreatedDate" : "2021-07-21T04:13:15.000+0000",
      "CreatedById" : "005xx000000HwDoIAK",
      "LastModifiedDate" : "2021-07-21T04:13:15.000+0000",
      "LastModifiedById" : "005xx000000HwDoIAK",
      "SystemModstamp" : "2021-07-21T04:13:15.000+0000",
      "LastViewedDate" : "2021-07-21T04:15:21.000+0000",
      "LastReferencedDate" : "2021-07-21T04:15:21.000+0000",
      "StartDate" : "2021-07-20",
      "IsActive" : true,
      "LoyaltyPromotionType" : "Cumulative",
      "IsEnrollmentRequired" : false,
      "CumulativeUsageTarget" : 100,
      "FulfillmentAction" : "CreditPoints",
      "LoyaltyProgramId" : "01pxx00000000HWHIAY",
      "TotalRewardPoints" : 100,
      "LoyaltyProgramCurrencyId" : "01cxx00000000XeIAI",
      "TransactionJournalType" : "Accrual",
      "IsPointsWithFactor" : false
    } ]
  }
} ]
```

## Get Tier

Retrieve the current tier for a specified loyalty program member.

This action is available in API version 51.0 and later for users with Loyalty Management license.



## Supported REST HTTP Methods

### URI

/services/data/v**XX.X**/actions/standard/getTier

### Formats

JSON, XML

### HTTP Methods

POST

### Authentication

Authorization: Bearer *token*

## Inputs

Input	Details
LoyaltyProgramMemberId	<p><b>Type</b> string</p> <p><b>Description</b> Required. ID of the loyalty program member to whom the tier details needs to be fetched.</p>
ProgramName	<p><b>Type</b> string</p> <p><b>Description</b> Required. The loyalty program name associated with the tier.</p>
TierGroupName	<p><b>Type</b> string</p> <p><b>Description</b> Required. The name of the tier group.</p>

## Outputs

Output	Details
LoyaltyProgramMemberId	<p><b>Type</b> string</p> <p><b>Description</b> The ID of the loyalty program member.</p>
SequenceNumber	<p><b>Type</b> integer</p> <p><b>Description</b> The ranking of the tier.</p>

Output	Details
TierGroupName	<p><b>Type</b> string</p> <p><b>Description</b> The name of the tier group.</p>
TierName	<p><b>Type</b> string</p> <p><b>Description</b> The name of the loyalty tier associated with the program member.</p>

## Example

### Sample Request

```
{
  "inputs": [{
    "TierGroupName": "QualifyingGroup",
    "ProgramName": "baLoyalty",
    "LoyaltyProgramMemberId": "01cxx00004000001AAA"
  }]
}
```

### Sample Response

```
{
  "errors": null,
  "isSuccess": true,
  "outputValues": {
    "LoyaltyProgramMemberId": "01Mxx0000A000001EAA",
    "TierGroupName": "QualifyingGroup",
    "TierName": "Gold",
    "ProgramName": "baLoyalty",
    "SequenceNumber": 4
  }
}
```

## Issue Voucher

Issue a voucher for a member or contact.

This action is available in API version 51.0 and later for users with Loyalty Management license.

## Supported REST HTTP Methods

### URI

/services/data/v**XX.X**/actions/standard/issueVoucher

**Formats**


JSON, XML

**HTTP Methods**

POST

**Authentication**Authorization: Bearer *token*

## Inputs

Input	Details
BenefitId	<p><b>Type</b> ID</p> <p><b>Description</b> The benefit associated with the voucher.</p>
ContactId	<p><b>Type</b> ID</p> <p><b>Description</b> The contact associated with the member.</p> <p> <b>Note:</b> Provide only <code>contactId</code> to retrieve vouchers issued to a Person Account. If both <code>accountId</code> and <code>contactId</code> are provided, then an error is displayed.</p>
DiscountPercent	<p><b>Type</b> percent</p> <p><b>Description</b> The discount percentage provided by the voucher. Specify a value to override the discount percentage specified in the associated voucher definition. Specify this when the associated voucher definition is of the type Discount Percentage. This field is available in API version 53.0 and later.</p>
FaceValue	<p><b>Type</b> double</p> <p><b>Description</b> The amount of discount provided by the voucher. Specify a value to override the face value specified in the associated voucher definition. Specify this when the associated voucher definition is of the type Fixed Value. This field is available in API version 53.0 and later.</p>
EffectiveDate	<p><b>Type</b> Date</p> <p><b>Description</b> The date the voucher goes into effect.</p>

Input	Details
ExpiryDate	<p><b>Type</b> Date</p> <p><b>Description</b> The date the voucher expires.</p>
JournalId	<p><b>Type</b> ID</p> <p><b>Description</b> The transaction journal associated with the member voucher when the voucher results from accrual of points.</p>
LoyaltyProgramMemberId	<p><b>Type</b> ID</p> <p><b>Description</b> ID of the loyalty program member to issue the voucher.</p>
PromotionId	<p><b>Type</b> ID</p> <p><b>Description</b> The promotion associated with the voucher.</p>
ProgramName	<p><b>Type</b> string</p> <p><b>Description</b> The name of the loyalty program associated with this voucher.</p>
Reason	<p><b>Type</b> string</p> <p><b>Description</b> The reason why the voucher is issued.</p>
VoucherCode	<p><b>Type</b> string</p> <p><b>Description</b> Required. The voucher identifier generated by an external system.</p>
VoucherDefinitionName	<p><b>Type</b> string</p> <p><b>Description</b> Required. The name of the voucher definition associated with the voucher code.</p>

## Outputs

Output	Details
VoucherId	<p><b>Type</b> ID</p> <p><b>Description</b> The ID of the voucher that's generated after the request is successful.</p>

## Example

### Sample Request

```
{
  "inputs": [{
    "LoyaltyProgramMemberId": "01Mxx0000A000001EAA",
    "ContactId": "01Mxx0000A000001EAA",
    "ProgramName": "Program 1",
    "VoucherDefinitionName": "Voucher1",
    "JournalId": "01Vxx0000A000001EAA",
    "VoucherCode": "Voucher123",
    "PromotionId": "01Mxx0000A000001EAA",
    "BenefitId": "01Mxx0000A000001EAA",
    "Reason": "Bonus",
    "EffectiveDate": "12-08-2021",
    "ExpiryDate": "12-09-2021"
    "DiscountPercent": "20"
  }]
}
```

### Sample Request to Retrieve Vouchers Issued to Accounts

```
{
  "inputs": [
    {
      "LoyaltyProgramMemberId": "01Mxx00000002HxEAI",
      "ContactId": "01Mxx0000A000001EAA",
      "AccountId": "001xx000003H3NnAAK",
      "VoucherDefinitionName": "Voucher1",
      "VoucherCode": "Voucher123",
      "ProgramName": "Sip & Save Rewards"
    }
  ]
}
```

### Sample Response

```
{
  "errors": null,
  "isSuccess": true,
  "outputValues": {
    "VoucherId": "01Yxx0000A000001EAA"
  }
}
```

```
}
}
```

## Merge Memberships Action

Merges two active loyalty program member records that belong to the same loyalty program. Out of the two memberships, one is source membership and the other is the target membership. After the merge is complete, the status of the source membership is changed to Merged and the status of the target membership remains active.

By default, all the transaction journals and points of the source member are transferred to the target member after the merge. Optionally, you can:

- Transfer all the vouchers of source member to the target member.
- Assign the higher current tier between the source and target members to the target member.

For more information about how the `mergeLoyaltyProgramMembership` action merges loyalty program memberships, see [Merge Memberships](#) in Salesforce Help.

This action is available in API version 56.0 and later for users in orgs where Loyalty Management is enabled.

## Supported REST HTTP Methods

### URI

`/services/data/vxx.x/actions/standard/mergeLoyaltyProgramMembership`

### Formats

JSON, XML

### HTTP Methods

POST

### Authentication

Authorization: Bearer *token*

## Inputs

Input	Details
<code>reasonForMergingMemberships</code>	<p><b>Type</b> String</p> <p><b>Description</b> Required. The reason why the source loyalty program member is to be merged with the target loyalty program member.</p>
<code>shouldHigherCurrentTierOfTargetMember</code>	<p><b>Type</b> Boolean</p> <p><b>Description</b> Indicates whether the tier for the target record is set to the source record's tier when the source tier is higher than the target record's current tier across all groups (<code>true</code>) or not (<code>false</code>).</p>

Input	Details
	The default value is <code>false</code> .
<code>shouldTransferVouchers</code>	<p><b>Type</b> Boolean</p> <p><b>Description</b> Indicates whether all vouchers that belong to the source loyalty program member must be transferred to the target loyalty program member (<code>true</code>) or not (<code>false</code>).  The default value is <code>false</code>.</p>
<code>sourceLoyaltyProgramMemberId</code>	<p><b>Type</b> ID</p> <p><b>Description</b> Required.  The ID of the loyalty program member record to be merged with the target loyalty program member record. The status of the source loyalty program member record is set to <code>Merged</code> after the merge is complete.</p>
<code>targetLoyaltyProgramMemberId</code>	<p><b>Type</b> ID</p> <p><b>Description</b> Required.  The ID of the loyalty program member record to remain active for the member after the merge is complete. The qualifying and non-qualifying points accrued by the source member are added to the respective points balances of the target member record.</p>

## Outputs

Output	Details
<code>errors</code>	<p><b>Type</b> <code>List&lt;String&gt;</code></p> <p><b>Description</b> A list of error messages if the operation fails.</p>
<code>isSuccess</code>	<p><b>Type</b> Boolean</p> <p><b>Description</b> Indicates whether the operation is successful or not.</p>

## Examples

### Sample Request

```
{
  "inputs": [ {
    "reasonForMergingMemberships": "Test Merge",
    "shouldSetHigherCurrentTierForTargetMember": true,
    "shouldTransferVouchers": true,
    "sourceLoyaltyProgramMemberId": "01MT100000001JnMAK",
    "targetLoyaltyProgramMemberId": "01MT100000001JnMAI"
  } ]
}
```

### Sample Response

```
[ {
  "actionName" : "mergeLoyaltyProgramMembership",
  "errors" : null,
  "isSuccess" : true
} ]
```

SEE ALSO:

[SOAP API Developer Guide: Core Data Types Used in API Calls](#)

## Process Member Benefit

---

Process the benefit action associated with the benefit that is assigned to a loyalty program member.

This action is available in API version 51.0 and later for users with Loyalty Management license.

## Supported REST HTTP Methods

### URI

/services/data/v**xx.x**/actions/standard/processMemberBenefitAction

### Formats

JSON, XML

### HTTP Methods

POST

### Authentication

Authorization: Bearer *token*

## Inputs

Input	Details
MemberBenefitId	Type ID



**Input****Details****Description**

Required. ID of the member associated with the benefit.

## Example

**Sample Request**

```
{
  "inputs": [{
    "MemberBenefitId": "01Mxx0000A000001EAA"
  }]
}
```

**Sample Response**

```
{
  "actionName": "executeMemberBenefit",
  "errors": null,
  "isSuccess": true,
}
```

## Redemption Cancellation

Revert a specific set of redemption transactions.

This action is available in API version 51.0 and later for users with Loyalty Management license.

## Supported REST HTTP Methods

**URI**

`/services/data/vXX.X/actions/standard/cancelRedemption`

**Formats**

JSON, XML

**HTTP Methods**

POST

**Authentication**

Authorization: Bearer *token*

## Inputs

**Input****Details**

CancellationReason

**Type**

string

**Input****Details**

	<b>Description</b>
	Required. The reason for cancellation of redemption transaction.
ExpiryDate	<b>Type</b> date  <b>Description</b> The date of expiration.
ReversalCategory	<b>Type</b> string  <b>Description</b> The category of reversal.
TransactionJournalId	<b>Type</b> string  <b>Description</b> Required. ID of the transaction journal.

## Outputs

**Output****Details**

	<b>Description</b>
CancellationTransactionJournalId	<b>Type</b> string  <b>Description</b> The ID of the cancellation transaction journal generated after the request is successful.

## Example

**Sample Request**

```
{
  "inputs": [{
    "TransactionJournalId": "0lvxx00004000001AAA",
    "CancellationReason": "Order Returned by Member",
    "ExpiryDate": "2020-04-15"
  }]
}
```

**Sample Response**

```
{
  "errors": null,
  "isSuccess": true,
}
```

```

"outputValues":{
  "CancellationTransactionJournalId":"01vxx00004000001AAB" }
}

```

## Run Program Process for Transaction Journal

---

Run active loyalty program processes for pending Accrual or Redemption type transaction journals.

The action assesses transaction journals and runs the program processes whose selected journal type matches the transaction's journal type. If program processes have a journal subtype selected, then processes only run for a transaction journal when the transaction journal's journal subtype matches that processes' journal subtype.

For more information about this action, see [Run Program Process For Transaction Journal](#) in Salesforce Help.

This action is available in API version 54.0 and later for users in org's where either B2C - Loyalty, B2C - Loyalty Plus, Loyalty Management - Growth, or Loyalty Management - Advanced license are enabled.

## Supported REST HTTP Methods

### URI

`/services/data/vXX.X/actions/standard/runProgramProcessForTransactionJournal`

### Formats

JSON, XML

### HTTP Methods

POST

### Authentication

Authorization: Bearer *token*

## Inputs

Input	Details
transactionJournalId	<p><b>Type</b> ID</p> <p><b>Description</b> Required. ID of the transaction journal for which program processes must be run. The status of the Transaction Journal must be Pending, and the journal type must be specified for it.</p>

## Outputs

Output	Details
status	<p><b>Type</b> Boolean</p>

**Output****Details****Description**

Indicates the status of the request. `True` indicates the program process was successfully run for the transaction journal.

## Example

**Sample Request**

```
{
  "inputs": [
    {
      "transactionJournalId": "01Mxx0000A000001EAA"
    }
  ]
}
```

**Sample Response**

```
[ {
  "actionName" : "runProgramProcessForTransactionJournal",
  "errors" : null,
  "isSuccess" : true,
  "outputValues":{
    "status":"true"
  }
} ]
```

## Transfer Member Points to Groups

Transfers Individual or Corporate type members' points contributions to the members' associated groups.

For more information about this action, see [Transfer Member Points to Groups](#) in Salesforce Help.

This action is available in API version 53.0 and later for users with Loyalty Management license.

## Supported REST HTTP Methods

**URI**

`/services/data/vXX.X/actions/standard/transferMemberPointsToGroups`

**Formats**

JSON, XML

**HTTP Methods**

POST

**Authentication**

Authorization: Bearer *token*

## Inputs

Input	Details
JournalId	<p><b>Type</b> ID</p> <p><b>Description</b> Required. The transaction journal from which points must be transferred to groups. The transaction journal must be associated to either an Individual or a Corporate type member. The points are transferred to the member's associated groups.</p>

## Example

### Sample Request

```
{
  "inputs": [
    {
      "JournalId": "01Mxx0000A000001EAA"
    }
  ]
}
```

### Sample Response

```
{
  "actionName": "transferMemberPointsToGroups",
  "errors": null,
  "isSuccess": true
}
```

## Unmerge Memberships Action

Unmerges loyalty program member records have a Merged status. The action unmerges memberships that are in the Merged status from the membership that it was previously merged with. After the unmerge is complete, the status of the Loyalty Program Member Merge record that was created during the merge is changed to Unmerged. The members are reassigned to the tier that was assigned to them before the merge. They are also credited with the points and transferred the vouchers they had before the membership was merged.

For more information about how the `unmergeLoyaltyProgramMembership` action unmerges loyalty program memberships, see [Unmerge Memberships](#) in Salesforce Help.

This action is available in API version 56.0 and later for users in orgs where Loyalty Management is enabled.

## Supported REST HTTP Methods

### URI

```
/services/data/vxx.x/actions/standard/unmergeLoyaltyProgramMembership
```

**Formats**

JSON, XML

**HTTP Methods**

POST

**Authentication**Authorization: Bearer *token*

## Inputs

Input	Details
<code>mergedLoyaltyProgramMemberId</code>	<b>Type</b> ID <b>Description</b> Required. The ID of the merged loyalty program member record to be unmerged.

## Outputs

Output	Details
<code>errors</code>	<b>Type</b> List<String> <b>Description</b> A list of error messages if the operation fails.
<code>isSuccess</code>	<b>Type</b> Boolean <b>Description</b> Indicates whether the operation is successful or not.

## Examples

**Sample Request**

```
{
  "inputs": [ {
    "mergedLoyaltyProgramMemberId": "01MT10000001JnMAK"
  } ]
}
```

**Sample Response**

```
[ {
  "actionName" : "unmergeLoyaltyProgramMembership",
  "errors" : null,
  "isSuccess" : true
} ]
```

## SEE ALSO:

[SOAP API Developer Guide: Core Data Types Used in API Calls](#)

## Update Member Progress for Cumulative Promotion

---

Update the progress made by a loyalty program member towards attaining a Cumulative type promotion. The action creates a Loyalty Program Member Promotion record if the promotion isn't associated with the member.

For more information about this action, see [Update the Member's Progress Towards Attaining a Cumulative Promotion](#) in Salesforce Help.

This action is available in API version 53.0 and later for users in orgs where either B2C - Loyalty, B2C - Loyalty Plus, Loyalty Management - Growth, or Loyalty Management - Advanced license is enabled.

## Supported REST HTTP Methods

**URI**

`/services/data/v $xx.x$ /actions/standard/updateProgressForCumulativePromotionUsage`

**Formats**

JSON, XML

**HTTP Methods**

POST

**Authentication**

Authorization: Bearer *token*

## Inputs

Input	Details
loyaltyProgramMemberId	<p><b>Type</b> ID</p> <p><b>Description</b> Required. ID of the loyalty program member who progress towards attaining a cumulative promotion must be updated.</p>
promotionId	<p><b>Type</b> ID</p>

**Input****Details****Description**

Required. ID of the Cumulative type promotion for which member's progress must be updated.

`usageIncrementForCumulativePromotion`

**Type**

integer

**Description**

Required. The value or quantity that must be added to member's usage towards attaining the promotion.

For example, if the target of a promotion is 10 units and the member has previously purchased 1 unit. Now the member has purchased 4 units. In this scenario, the input must be 4 and not 5 (1+4).

## Outputs

**Output****Details**

`cumulativeUsageTarget`

**Type**

integer

**Description**

The target value or quantity which members must achieve to attain the promotion.

`updatedCumulativeUsageCompleted`

**Type**

integer

**Description**

The updated value or quantity by which the member has progressed towards attaining the promotion.

For example, if the target of a promotion is 10 units and the member has previously purchased 1 unit. Now the member has purchased 4 units. In this scenario, the output is 5.

## Example

**Sample Request**

```
{
  "inputs" : [
    {
      "loyaltyProgramMemberId" : "01MRxx0000006MfMAI",
      "promotionId": "0c8xx00000002MqIAI",
      "usageIncrementForCumulativePromotion": 4
    }
  ]
}
```



**Sample Response**

```
[ {  
  "actionName" : "updateProgressForCumulativePromotionUsage",  
  "errors" : null,  
  "isSuccess" : true,  
  "outputValues" : {  
    "cumulativeUsageTarget" : 10,  
    "updatedCumulativeUsageCompleted" : 5  
  }  
} ]
```

## CHAPTER 8 Loyalty Management Custom Invocable Actions

In this chapter ...

- [Run Program Process](#)

Create and manage loyalty programs for your organization using custom invocable actions.

## Run Program Process

---

This action lists custom actions associated with all the available loyalty program processes. You can trigger an active loyalty program process using its corresponding custom invocable action.

This action is available in API version 56.0 and later.

## Supported REST HTTP Methods

### URI

`/services/data/v56.0/actions/custom/runProgramProcess/<actionName>`



**Note:** `<actionname>` is the name of the custom action associated with an active loyalty program process. The custom action name consists of the name of the loyalty program that the process belongs to followed by a delimiter, and then the name of the process. The delimiter is `SLI`. The program name, the delimiter, and the process name are separated by underscores. If your loyalty program and process names have spaces, the spaces are replaced with underscores.

If the name of your loyalty program is `InnerCircle` and process name is `Issue Voucher`, then the action name is `InnerCircle_SLI_Issue_Voucher`. If the name of your loyalty program is `Inner Circle` and process name is `IssueVoucher`, then the action name is `Inner_Circle_SLI_IssueVoucher`.

### Formats

JSON

### HTTP Methods

POST

### Authentication

Authorization: Bearer *token*

## Inputs

Vary depending on the Loyalty Program Process setup action.

## Outputs

Standard invocable action result parameters, such as `isSuccess` and `errors` along with other parameters that vary depending on the inputs.

Output	Details
<code>errors</code>	<p><b>Type</b> <code>List&lt;String&gt;</code></p> <p><b>Description</b> <b>A list of error messages if the operation fails.</b></p>
<code>isSuccess</code>	<p><b>Type</b> Boolean</p> <p><b>Description</b> Indicates whether the operation is successful or not.</p>

## Examples

This section has the sample request and response for triggering the Credit Points Program Process.

### Sample Resource

/services/data/v56.0/actions/custom/runProgramProcess/NTO\_SLI\_CreditPoints

### Sample request

Here's an example POST request that has the inputs, such as, Loyalty Program member ID, points to credit, and other transaction details:

```
{
  "inputs": [
    {
      "TransactionJournal": {
        "ActivityDate": "2022-11-27T00:00:00Z",
        "BookingDate": "2022-10-23",
        "EndDate": "2022-10-23T00:00:00Z",
        "ExternalTransactionNumber": "1",
        "InvoiceDate": "2022-10-23",
        "JournalTypeName": "Accrual",
        "JournalSubTypeName": "xyz",
        "LoyaltyProgramId": "01pRM0000004D53YAE",
        "MemberId": "01MRM0000004Eqv2AE",
        "TransactionAmount": 45,
        "Status": "Processed"
      },
      "PointsToCredit": "100"
    }
  ]
}
```

### Sample response

Here's an example response of a successful credit points program operation:

```
{
  "actionName": "NTO_SLI_CreditPoints",
  "errors": null,
  "isSuccess": true,
  "outputValues": {
    "PointsCredited": 100
  }
}
```

## CHAPTER 9 Loyalty Management Metadata API Types

### In this chapter ...

- [BenefitAction](#)
- [LoyaltyProgramSetup](#)
- [Settings](#)

Metadata API enables you to access some types and feature settings that you can customize in the user interface. For more information about Metadata API and to find a complete reference of existing metadata types, see [Metadata API Developer Guide](#).

## BenefitAction

---

Represents details of an action that can be triggered automatically for a benefit. This type extends the Metadata metadata type and inherits its `fullName` field.

 **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.

### File Suffix and Directory Location

BenefitAction components have the suffix `.benefitAction` and are stored in the `benefitActions` folder.

### Version

BenefitAction components are available in API version 52.0 and later.

### Special Access Rules

To use this metadata type, the Loyalty Management license is required

### Fields

Field Name	Field Type	Description
<code>benefitActionParameters</code>	<a href="#">BenefitActionParameter</a> on page 495[]	The collection of flow input parameters to be used in a benefit action.
<code>description</code>	string	The description of the benefit action.
<code>flowDefinition</code>	string	The ID of the flow associated with the benefit action.
<code>isActive</code>	boolean	Required. Indicates whether the benefit action is active (true) or inactive (false). The default value is <code>false</code> .
<code>isBenefitExpirationAllowed</code>	boolean	Indicates whether the benefit action is allowed to expire a benefit (true) or not (false). The default value is <code>false</code> .
<code>isBenefitUpdateAllowed</code>	boolean	Indicates whether the benefit action is allowed to update a benefit (true) or not (false). The default value is <code>false</code> .
<code>masterLabel</code>	string	Required. The user interface name of the benefit action.
<code>memberBenefitVariableName</code>	string	The name of the input variable that stores the member benefit ID in the flow specified for the benefit action.
<code>processType</code>	string	Required. The process type of the benefit action. The default value is <code>Loyalty</code> .
<code>type</code>	string	Required. The type of the benefit action.

## BenefitActionParameter

Field Name	Field Type	Description
dataType	BenefitActionParameter of type string)	Required. The data type of the parameter. Possible values are: <ul style="list-style-type: none"> <li>• Boolean</li> <li>• Date</li> <li>• DateTime</li> <li>• EntityPicklist</li> <li>• Number</li> <li>• Picklist</li> <li>• String</li> </ul>
masterLabel	string	Required. The user interface name of the parameter.
name	string	The name of the parameter.
objectName	string	The object associated with the parameter when the data type is record.
parameterValues	<a href="#">BenefitActionParameterValue</a> on page 495[]	A collection of predefined values for the benefit action parameter.
picklistName	string	The name of the picklist fields in the object associated with the parameter.

## BenefitActionParameterValue

Field Name	Field Type	Description
parameterValue	string	Required. The predefined value of the parameter.

## Declarative Metadata Sample Definition

The following is an example of a BenefitAction component.

```
<?xml version="1.0" encoding="UTF-8"?>
<BenefitAction xmlns="http://soap.sforce.com/2006/04/metadata">
  <description>Metadata API Test</description>
  <flowDefinition>test</flowDefinition>
  <isActive>true</isActive>
  <isBenefitExpirationAllowed>true</isBenefitExpirationAllowed>
  <isBenefitUpdateAllowed>true</isBenefitUpdateAllowed>
  <masterLabel>Test1</masterLabel>
  <processType>1</processType>
  <memberBenefitVariableName>test</memberBenefitVariableName>
  <benefitActionParameters>
    <name>test</name>
    <picklistName>Type</picklistName>
  </benefitActionParameters>
</BenefitAction>
```

```

    <masterLabel>test</masterLabel>
    <dataType>Picklist</dataType>
    <objectName>Account</objectName>
    <parameterValues>
      <parameterValue>string</parameterValue>
    </parameterValues>
  </benefitActionParameters>
  <type>1</type>
</BenefitAction>

```

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>BenefitAction</name>
  </types>
  <types>
    <members>test</members>
    <name>Flow</name>
  </types>
  <version>52.0</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](#).

## LoyaltyProgramSetup

---

Represents the configuration of a loyalty program process including its parameters and rules. Program processes determine how new transaction journals are processed. When new transaction journals meet the criteria and conditions for a program process, actions that are set up in the process are triggered for the transaction journals.

 **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

LoyaltyProgramSetup components have the suffix `loyaltyProgramSetup` and are stored in the `loyaltyProgramSetups` folder.



## Version

LoyaltyProgramSetup components are available in API version 54.0 and later for Loyalty Management and in API version 59.0 and later for Referral Marketing.

## Special Access Rules

To use this metadata type, your org must have either B2C - Loyalty, B2C - Loyalty Plus, Loyalty Management - Growth, Loyalty Management - Advanced, or Referral Marketing license enabled.

## Fields


Field Name	Description
label	<p><b>Field Type</b> string</p> <p><b>Description</b> Name of the loyalty program that the program process is associated with. If a loyalty program or referral program with the specified name doesn't exist, a new LoyaltyProgram record is created. The name of a program must contain at least one alphanumeric character.</p>
programProcesses	<p><b>Field Type</b> <a href="#">LoyaltyProgramProcess[]</a> on page 497</p> <p><b>Description</b> Collection of loyalty program processes associated with a loyalty program or a referral program.</p>

## LoyaltyProgramProcess

Represents a collection of fields relating to a loyalty program process.

Field Name	Description
description	<p><b>Field Type</b> string</p> <p><b>Description</b> The description of the loyalty program process.</p>
executionType	<p><b>Field Type</b> LoyaltyPgmProcExecutionType (enumeration of type string)</p> <p><b>Description</b> The mode of processing transaction journals by the loyalty program process. Possible values are:</p>

Field Name	Description
	<ul style="list-style-type: none"> <li>• Batch</li> <li>• BatchAndRealTime</li> <li>• RealTime</li> </ul>
journalSubType	<p><b>Field Type</b> string</p> <p><b>Description</b> The subtype of transaction journals processed by the loyalty program process.</p>
journalType	<p><b>Field Type</b> string</p> <p><b>Description</b> The type of transaction journal processed by the loyalty program process. Possible values for loyalty program:</p> <ul style="list-style-type: none"> <li>• Accrual</li> <li>• Redemption</li> </ul> <p>Possible value for referral program:</p> <ul style="list-style-type: none"> <li>• Referral</li> </ul>
loyaltyTierGroup	<p><b>Field Type</b> string</p> <p><b>Description</b> The tier group of a loyalty program. This field is available in API version 56.0 and later. This field isn't applicable for referral programs.</p>
parameters	<p><b>Field Type</b> <a href="#">LoyaltyProgramProcessParameter[]</a> on page 499</p> <p><b>Description</b> The parameters associated with the loyalty program process.</p>
processName	<p><b>Field Type</b> string</p> <p><b>Description</b> Required. The name of the loyalty program process.</p>
processType	<p><b>Field Type</b> string</p> <p><b>Description</b> Required.</p>

Field Name	Description
	The type of records processed by the loyalty program process. For referral programs, the process type is <code>TransactionJournal</code> .
<code>rules</code>	<p><b>Field Type</b>  <a href="#">LoyaltyProgramProcessRule[]</a> on page 503</p> <p><b>Description</b>            The rules associated with the loyalty program process.</p>
<code>status</code>	<p><b>Field Type</b>            LoyaltyPgmProcStatus (enumeration of type string)</p> <p><b>Description</b>            The status of the loyalty program process.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>• Active</li> <li>• Draft</li> <li>• Inactive</li> </ul> <p> <b>Note:</b> Only active program processes can process transaction journals.</p>

## LoyaltyProgramProcessParameter

Represents a collection of fields relating to a parameter that's associated with the program process. Parameters are dynamic or fixed values that are used in rule. You can define the value of a parameter based on its type and data type.

Field Name	Description
<code>condition</code>	<p><b>Field Type</b>  <a href="#">LoyaltyProgramProcessCondition</a> on page 501</p> <p><b>Description</b>            The filter condition that decides which records are stored in the parameter.</p>
<code>dataType</code>	<p><b>Field Type</b>            LoyaltyPgmProcParmDataType (enumeration of type string)</p> <p><b>Description</b>            The data type of the parameter. Determines the type of value that can be stored in the parameter.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>• Boolean</li> <li>• Date</li> <li>• DateTime</li> <li>• Numeric</li> </ul>

Field Name	Description
	<ul style="list-style-type: none"> <li>• Subject</li> <li>• Text</li> </ul>
decimalPlaces	<p><b>Field Type</b> int</p> <p><b>Description</b> The number of decimal places supported by the parameter when it is of the type Variable and data type Numeric.</p>
description	<p><b>Field Type</b> string</p> <p><b>Description</b> The description of the parameter.</p>
isCollection	<p><b>Field Type</b> boolean</p> <p><b>Description</b> Indicates whether the parameter can store multiple values when it is of the type Variable.</p>
isInput	<p><b>Field Type</b> boolean</p> <p><b>Description</b> Indicates whether a parameter can be used as an input outside the loyalty program process.</p>
isOutput	<p><b>Field Type</b> boolean</p> <p><b>Description</b> Indicates whether a parameter can be used as an output outside the loyalty program process.</p>
objectName	<p><b>Field Type</b> string</p> <p><b>Description</b> Name of the object whose records are stored by the parameter when it is of the type Variable and data type sObject.</p>
parameterName	<p><b>Field Type</b> string</p> <p><b>Description</b> Required.</p>

Field Name	Description
	The name of the parameter.
parameterType	<p><b>Field Type</b> LoyaltyPgmProcParmType (enumeration of type string)</p> <p><b>Description</b> The type of value the parameter can store. Possible values are:</p> <ul style="list-style-type: none"> <li>• Constant</li> <li>• Formula</li> <li>• Variable</li> </ul>
value	<p><b>Field Type</b> string</p> <p><b>Description</b> The value of the parameter when it is of the type Variable or Formula and isn't of the data type sObject.</p>

## LoyaltyProgramProcessCondition

Represents a collection of fields relating to a condition. Conditions filter records that parameters store or check whether child actions must be triggered for a transaction journal.

Field Name	Description
conditionCriteria	<p><b>Field Type</b> string</p> <p><b>Description</b> Required. The criteria that determine when the condition is met by a record or by a transaction journal.</p>
conditionFilterCriteria	<p><b>Field Type</b> <a href="#">LoyaltyProgramProcessConditionFilterCriteria[]</a> on page 502</p> <p><b>Description</b> The filter criteria that determines which records or transaction journals are filtered.</p>
conditionName	<p><b>Field Type</b> string</p> <p><b>Description</b> Required. The name of the condition.</p>

## LoyaltyProgramProcessConditionFilterCriteria

Represents a collection of fields relating to a filter criteria that's part of a condition. Multiple filter criteria can be added for a condition. Filter criteria determine which records are filtered by related condition.

Field Name	Description
operator	<p><b>Field Type</b> LoyaltyPgmProcCondOperator (enumeration of type string)</p> <p><b>Description</b> Required. The operator of the filter criteria. Possible values are:</p> <ul style="list-style-type: none"><li>• Contains</li><li>• DoesNotContain</li><li>• EndsWith</li><li>• Equals</li><li>• GreaterThan</li><li>• GreaterThanOrEquals</li><li>• IsNotNull</li><li>• IsNull</li><li>• LessThan</li><li>• LessThanOrEquals</li><li>• NotEquals</li><li>• StartsWith</li></ul>
sequence	<p><b>Field Type</b> int</p> <p><b>Description</b> Required. The sequence number of the filter criteria within a condition.</p>
sourceFieldName	<p><b>Field Type</b> string</p> <p><b>Description</b> Required. The name of the field used in the filter criteria.</p>
value	<p><b>Field Type</b> string</p> <p><b>Description</b> The value of the filter criteria.</p>

Field Name	Description
valueType	<p><b>Field Type</b> LoyaltyPgmProcCondType (enumeration of type string)</p> <p><b>Description</b> Required. The type of value specified in the filter criteria. Possible values are:</p> <ul style="list-style-type: none"> <li>• Formula</li> <li>• Literal</li> <li>• Lookup</li> <li>• Parameter</li> </ul>

## LoyaltyProgramProcessRule

Represents a collection of fields relating to a rule. A rule consists of a set of conditions and actions.

Field Name	Description
actions	<p><b>Field Type</b> <a href="#">LoyaltyProgramProcessAction[]</a> on page 504</p> <p><b>Description</b> The actions associated with the rule.</p>
conditions	<p><b>Field Type</b> <a href="#">LoyaltyProgramProcessCondition[]</a> on page 501</p> <p><b>Description</b> The conditions associated with the rule.</p>
description	<p><b>Field Type</b> string</p> <p><b>Description</b> The description of the rule.</p>
endDate	<p><b>Field Type</b> date</p> <p><b>Description</b> The date until which the rule processes transaction journals.</p>
previousRule	<p><b>Field Type</b> string</p>

Field Name	Description
	<p><b>Description</b></p> <p>The rule that processes new transaction journals before the current rule. The current rule is triggered when the previous rule completes processing transaction journals.</p>
promotion	<p><b>Field Type</b></p> <p>string</p> <p><b>Description</b></p> <p>The promotion associated with the rule. When a promotion is associated with a rule, the start date, end date, and status of the promotion determines the corresponding fields of the rule.</p>
ruleName	<p><b>Field Type</b></p> <p>string</p> <p><b>Description</b></p> <p>Required.</p> <p>The name of the rule.</p>
startDate	<p><b>Field Type</b></p> <p>date</p> <p><b>Description</b></p> <p>The date from which the rule starts processing transaction journals.</p>
status	<p><b>Field Type</b></p> <p>LoyaltyPgmProcRuleStatus (enumeration of type string)</p> <p><b>Description</b></p> <p>The status of the rule.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>• Active</li> <li>• Draft</li> <li>• Inactive</li> </ul>
stepMappings	<p><b>Field Type</b></p> <p><a href="#">LoyaltyProgramProcessRuleStepMapping[]</a> on page 509</p> <p><b>Description</b></p> <p>The list of step mappings associated with rule.</p>




## LoyaltyProgramProcessAction

Represents a collection of fields relating to an action.



Field Name	Description
actionName	<p><b>Field Type</b> string</p> <p><b>Description</b> Required. The name of the action.</p>
actionParameters	<p><b>Field Type</b> <a href="#">LoyaltyProgramProcessActionParameter[]</a> on page 508</p> <p><b>Description</b> The parameters of the action.</p>
actionType	<p><b>Field Type</b> LoyaltyPgmProcActionType (enumeration of type string)</p> <p><b>Description</b> Required. The type of action. Possible values are:</p> <ul style="list-style-type: none"> <li>• These values are used in Loyalty Management: <ul style="list-style-type: none"> <li>– AssignParameterValues—Assigns values to parameters.</li> <li>– AssignBadgeToMember—Assigns a badge to a loyalty program member. This value is available in API version 56.0 and later.</li> <li>– Crud—Creates or updates records in the target object. This value is available in API version 56.0 and later.</li> <li>– CheckMemberBadgeAssignment—Checks whether a badge is assigned to a loyalty program member. This value is available in API version 56.0 and later.</li> <li>– ChangeMemberTier—Changes the tier of a loyalty program member. This value is available in API version 56.0 and later.</li> <li>– CreditPoints—Credits points to the loyalty program member associated with the transaction journal that's processed by the rule.</li> <li>– DebitPoints—Debits points from the points balance of the loyalty program member associated with the transaction journal that's processed by the rule.</li> <li>– GetMemberAttributesValues—Gets the details of the loyalty program member's attribute value for the selected engagement attribute. This value is available in API version 55.0 and later.</li> <li>– GetMemberPointBalance—Gets the points balance of a loyalty program member.</li> <li>– GetMemberPromotions—Get promotions of a loyalty program member. This value is available in API version 56.0 and later.</li> <li>– GetMemberTier—Gets the tier details of a loyalty program member.</li> </ul> </li> </ul>

Field Name	Description
	<ul style="list-style-type: none"> <li>- <code>GetOutputsFromDecisionTable</code>—Gets outputs provided by a decision table. This value is available in API version 56.0 and later.</li> <li>- <code>—</code>—Adds the specified value to the loyalty program member's usage towards achieving a cumulative promotion by a specified value.</li> <li>- <code>IncreaseUsageForCumulativePromotion</code>—Increases a loyalty program member's usage for a cumulative promotion.</li> <li>- <code>IssueVoucher</code>—Issues a voucher to the loyalty program member associated with the transaction journal that's processed by the rule.</li> <li>- <code>RedeemVoucher</code>—Redeems a voucher for the loyalty program member associated with the transaction journal that's processed by the rule. This value is available in API version 58.0 and later.</li> <li>- <code>—</code>—Updates the loyalty program member's usage towards achieving a cumulative promotion by a specified value.</li> <li>- <code>RunFlow</code>—Runs a flow.</li> <li>- <code>RunProgramProcess</code>—Runs an active loyalty program process as a subprocess. This value is available in API version 56.0 and later.</li> <li>- <code>SendMail</code>—Sends emails to the loyalty program member for whom the process is run. This value is available in API version 59.0 and later.</li> <li>- <code>UpdateCurrentValueForMemberAttribute</code>—Updates the loyalty program member's current attribute value for the selected engagement attribute. This value is available in API version 55.0 and later.</li> <li>- <code>UpdatePointBalance</code>—Updates the points balance of the loyalty program member associated with the transaction journal that's processed by the rule.</li> <li>- <code>UpdateUsageForCumulativePromotion</code>—Updates a loyalty program member's usage for a cumulative promotion.</li> </ul> <ul style="list-style-type: none"> <li>• These values are used in Referral Marketing:           <ul style="list-style-type: none"> <li>- <code>AssignParameterValues</code>—Assigns values to parameters.</li> <li>- <code>Crud</code>—Creates or updates records in the target object.</li> <li>- <code>GetMemberAttributesValues</code>—Gets the details of an advocate's attribute value for the selected engagement attribute.</li> <li>- <code>GetMemberPromotions</code>—Gets the promotions of an advocate.</li> <li>- <code>GetOutputsFromDecisionTable</code>—Gets outputs provided by a decision table.</li> <li>- <code>IssueVoucher</code>—Issues a voucher to an advocate or a referred friend.</li> <li>- <code>RedeemVoucher</code>—Redeems a voucher for an advocate or a friend.</li> <li>- <code>SendMail</code>—Sends emails to a referral program's advocates and referrals.</li> <li>- <code>UpdateCurrentValueForMemberAttribute</code>—Updates an advocate's current attribute value for the selected engagement attribute.</li> </ul> </li> </ul>

Field Name	Description
<code>crudActionType</code>	<p><b>Field Type</b> LoyaltyPgmProcCrudActType (enumeration of type string)</p> <p><b>Description</b> The type of operation to perform on target object records by the action. This field is available from API version 56.0 and later.</p> <p> <b>Note:</b> This field is required when the <code>actionType</code> field value is <code>CRUD</code>.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>• <code>create</code></li> <li>• <code>update</code></li> </ul>
<code>decisionTable</code>	<p><b>Field Type</b> string</p> <p><b>Description</b> The decision that's invoked by the action for the transaction journal that's processed by the rule.</p>
<code>decisionTableDatasetLink</code>	<p><b>Field Type</b> string</p> <p><b>Description</b> The dataset link associated with the selected decision table.</p>
<code>entityApiName</code>	<p><b>Field Type</b> string</p> <p><b>Description</b> The API name of the target object. This field is available from API version 56.0 and later.</p> <p> <b>Note:</b> This field is required when the <code>actionType</code> field value is <code>CRUD</code>.</p>
<code>flowDefinition</code>	<p><b>Field Type</b> string</p> <p><b>Description</b> The flow that's run by the action for the transaction journal that's processed by the rule. The selected flow must be of the type <code>LoyaltyManagementFlow</code>.</p>
<code>loyaltyProgramProcess</code>	<p><b>Field Type</b> string</p> <p><b>Description</b> The subprogram processes that's run by the action. This field is available from API version 56.0 and later.</p> <p> <b>Note:</b> This field is required when the <code>actionType</code> field value is <code>RunProgramProcess</code>.</p>

## LoyaltyProgramProcessActionParameter

Represents a collection of fields relating to an action parameter. A parameter is either an input or an output for the action. Input parameters store the values used by the action. Output parameters store the result of the action.

Field Name	Description
<code>operator</code>	<p><b>Field Type</b> LoyaltyPgmProcActParamOper (enumeration of type string)</p> <p><b>Description</b> The type of operator used in the action. This field is available in API version 56.0 and later.</p> <p>Possible value is:</p> <ul style="list-style-type: none"> <li>• <code>Equals</code></li> </ul>
<code>parameterName</code>	<p><b>Field Type</b> string</p> <p><b>Description</b> Required.</p> <p>The name of parameter. The parameter name must be the same as the input or the output field that's supported depending on the associated action's type.</p>
<code>sequenceNumber</code>	<p><b>Field Type</b> int</p> <p><b>Description</b> The sequence number of the parameter in the action. This field is available in API version 56.0 and later.</p>
<code>value</code>	<p><b>Field Type</b> string</p> <p><b>Description</b> Required.</p> <p>The value of the parameter.</p>
<code>valueType</code>	<p><b>Field Type</b> LoyaltyPgmProcActParamType (enumeration of type string)</p> <p><b>Description</b> The type of value to provide in the parameter. This field is available in API version 56.0 and later.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>• <code>Literal</code>—A constant value.</li> <li>• <code>Parameter</code>—A runtime value passed using a parameter.</li> </ul>

## LoyaltyProgramProcessRuleStepMapping

Represents a collection of fields relating to a step mapping. Map conditions with child actions or map an action without a parent step.

Field Name	Description
associatedStep	<p><b>Field Type</b> string</p> <p><b>Description</b> Required. The action that's associated with the mapping.</p>
parentStep	<p><b>Field Type</b> string</p> <p><b>Description</b> The condition that contains one or more child actions.</p>
sequence	<p><b>Field Type</b> int</p> <p><b>Description</b> Required. The sequence number of the mapping within a rule.</p>

## Declarative Metadata Sample Definition

The following is an example of a LoyaltyProgramSetup component.

```
<?xml version="1.0" encoding="UTF-8"?>
<LoyaltyProgramSetup xmlns="http://soap.sforce.com/2006/04/metadata">
  <label>Cloud Kicks Inner Circle</label>
  <programProcesses>
    <executionType>RealTime</executionType>
    <parameters>
      <dataType>Numeric</dataType>
      <decimalPlaces>0</decimalPlaces>
      <isCollection>>false</isCollection>
      <isInput>>false</isInput>
      <isOutput>>false</isOutput>
      <parameterName>VoucherValue</parameterName>
      <parameterType>Constant</parameterType>
      <value>50</value>
    </parameters>
    <processName>Issue Vouchers</processName>
    <processType>Transaction Journal</processType>
    <rules>
      <actions>
        <actionName>Issue High Transaction Value Voucher</actionName>
      </actions>
    </rules>
  </programProcesses>
</LoyaltyProgramSetup>
```

```

    <actionParameters>
      <operator>Equals</operator>
      <parameterName>VoucherDefinitionName</parameterName>
      <sequenceNumber>1</sequenceNumber>
      <value>Voucher for High Value Transactions</value>
      <valueType>Literal</valueType>
    </actionParameters>
    <actionParameters>
      <operator>Equals</operator>
      <parameterName>VoucherCode</parameterName>
      <sequenceNumber>2</sequenceNumber>
      <value>{!TransactionJournal.Order.Id}</value>
    </actionParameters>
    <actionParameters>
      <operator>Equals</operator>
      <parameterName>VoucherEffectiveDate</parameterName>
      <sequenceNumber>3</sequenceNumber>
      <value>DATEVALUE (&quot;2021-11-21 00:00:00&quot;)</value>
    </actionParameters>
    <actionParameters>
      <operator>Equals</operator>
      <parameterName>VoucherExpirationDate</parameterName>
      <sequenceNumber>4</sequenceNumber>
      <value>DATEVALUE (&quot;2022-01-01 00:00:00&quot;)</value>
    </actionParameters>
    <actionParameters>
      <operator>Equals</operator>
      <parameterName>VoucherFaceValue</parameterName>
      <sequenceNumber>5</sequenceNumber>
      <value>{!VoucherValue}</value>
    </actionParameters>
    <actionType>IssueVoucher</actionType>
  </actions>
  <conditions>
    <conditionCriteria>1</conditionCriteria>
    <conditionFilterCriteria>
      <operator>GreaterThanOrEquals</operator>
      <sequence>1</sequence>
      <sourceFieldName>TransactionJournal.TransactionAmount</sourceFieldName>

      <value>100</value>
      <valueType>Literal</valueType>
    </conditionFilterCriteria>
    <conditionName>New Condition</conditionName>
    <conditionType>Condition</conditionType>
  </conditions>
  <endDate>2022-01-01</endDate>
  <ruleName>Issue Voucher for Transactions Above $100</ruleName>
  <startDate>2021-11-21</startDate>
  <status>Draft</status>
  <stepMappings>
    <associatedStep>New Condition</associatedStep>
    <sequence>1</sequence>
  </stepMappings>

```

```

        <stepMappings>
            <associatedStep>Issue High Transaction Value Voucher</associatedStep>
            <parentStep>New Condition</parentStep>
            <sequence>1</sequence>
        </stepMappings>
    </rules>
    <status>Draft</status>
</programProcesses>
</LoyaltyProgramSetup>

```

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

```

<?xml version="1.0" encoding="UTF-8"?>
<!--
  ~ Copyright 2020 Salesforce, Inc.
  ~ All Rights Reserved
  ~ Company Confidential
-->
<Package xmlns="http://soap.sforce.com/2006/04/metadata">
    <types>
        <members>*</members>
        <name>LoyaltyProgramSetup</name>
    </types>
    <version>54.0</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](#).

## Settings

---

Represents the organization settings to control the update of non-qualifying points balance for members when points are credited. For more information, see [Settings](#).

### [IndustriesGamificationSettings](#)

Represents the settings to enable games in Loyalty Management.

### [IndustriesLoyaltySettings](#)

Represents the settings to enable capabilities of Loyalty Management.

### [IndustriesUnifiedPromotionsSettings](#)

Represents the settings to Global Promotions Management.

## IndustriesGamificationSettings

Represents the settings to enable games in Loyalty Management.

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

In the package manifest, all organization settings metadata types are accessed using the Settings name. See [Settings](#) for more details.

## File Suffix and Directory Location

IndustriesGamificationSettings values are stored in a single file named `IndustriesGamification.settings` 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

IndustriesGamificationSettings components are available in API version 60.0 and later.

## Fields

Field Name	Field Type	Description
<code>enableGamification</code>	boolean	Indicates whether the capability to offer games to loyalty program members as program reward is enabled ( <code>true</code> ) or disabled ( <code>false</code> ) for your org. The default value is <code>false</code> .

## Declarative Metadata Sample Definition

The following is an example of a `IndustriesGamificationSettings` component.

```
<?xml version="1.0" encoding="UTF-8"?>
<IndustriesGamificationSettings xmlns="http://soap.sforce.com/2006/04/metadata">
  <enableGamification>true</enableGamification>
</IndustriesGamificationSettings>
```

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>IndustriesGamification</members>
    <name>Settings</name>
  </types>
  <version>60.0</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](#).

## IndustriesLoyaltySettings

Represents the settings to enable capabilities of Loyalty Management.

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



In the package manifest, all organization settings metadata types are accessed using the Settings name. See Settings for more details.

## File Suffix and Directory Location

IndustriesLoyaltySettings values are stored in a single file named `IndustriesLoyalty.settings` 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

IndustriesLoyaltySettings components are available in API version 53.0 and later.

## Fields

Field Name	Field Type	Description
<code>enableAutomaticVoucherCodeGeneration</code>	boolean	Indicates whether the capability to automatically generate voucher codes is enabled ( <code>true</code> ) or not ( <code>false</code> ). This field is available in API version 57.0 and later.
<code>enableNQPREalTimePointBalance</code>	boolean	Indicates whether the capability to update the non-qualifying point balance of members in real time is enabled ( <code>true</code> ) or disabled ( <code>false</code> ) for your org. The default value is <code>true</code> .
<code>enableFixedTypeNQPAggregation</code>	boolean	Indicates whether the capability to aggregate and expire fixed type non-qualifying points in batches is enabled ( <code>true</code> ) or disabled ( <code>false</code> ) for your org. The default value is <code>true</code> . This field is available in API version 54.0 or later.
<code>enableNonQualifyingPointsConsolidation</code>	boolean	Indicates whether the capability to aggregate and expire fixed type non-qualifying points in real time is enabled ( <code>true</code> ) or disabled ( <code>false</code> ) for your org. The default value is <code>true</code> . This field is available in API version 58.0 or later.
<code>enableQPREalTimePointBalance</code>	boolean	Indicates whether the real time qualifying points balance update capability is enabled ( <code>true</code> ) or disabled ( <code>false</code> ) for your org. The default value is <code>true</code> . This field is available in API version 55.0 or later.
<code>enableLoyaltyRulesVerifyCPMemberSegment</code>	boolean	Indicates whether the capability that allows: <ul style="list-style-type: none"> <li>• Loyalty program process rules to process transaction journals only when the loyalty program member is part of a Data Cloud segment associated with the rule's promotion is enabled (<code>true</code>) or disabled (<code>false</code>) for your org.</li> <li>• The Promotion Eligibility component on Loyalty Program Member record page to categorize promotions based on whether the member belongs to the promotion's campaign or Data Cloud segment is enabled (<code>true</code>) or disabled (<code>false</code>) for your org.</li> </ul> The default value is <code>false</code> . This field is available in API version 55.0 or later.

Field Name	Field Type	Description
<code>enableLoyaltyRedeemPointsExpirationInfoPref</code>	boolean	Indicates whether the capability that automatically calculates and adds the expiration date of points credited back to members for canceled redemptions is enabled ( <code>true</code> ) or disabled ( <code>false</code> ) for your org. The default value is <code>true</code> . This field is available in API version 55.0 or later.
<code>enableLoyaltyServiceExcellence</code>	boolean	Indicates whether Service Console for Loyalty Management is enabled ( <code>true</code> ) or disabled ( <code>false</code> ) for your org. The default value is <code>false</code> . This field is available in API version 57.0 or later.
<code>enableLoyaltyApiAccessForExternalSiteUsers</code>	boolean	Indicates whether the capability that allows customers to join and leave loyalty programs and allows members to join and opt out of promotions from Experience Cloud sites is enabled ( <code>true</code> ) or disabled ( <code>false</code> ) for your org. The default value is <code>false</code> . This field is available in API version 58.0 or later.
<code>enableAutomaticMemberTierAssessmentSelection</code>	boolean	Indicates whether the capability that automatically selects members as eligible for tier assessment when members' qualifying points balance changes is enabled ( <code>true</code> ) or disabled ( <code>false</code> ) for your org. The default value is <code>false</code> . This field is available in API version 58.0 or later.
<code>enablePointsLifecycleTracking</code>	boolean	Indicates whether the capability that allows the company to trace how members redeem their accrued fixed-model non-qualifying points is enabled ( <code>true</code> ) or disabled ( <code>false</code> ) for your org. The default value is <code>false</code> . This field is available in API version 62.0 or later.
<code>enableNegativePointBalance</code>	boolean	Indicates whether the capability that lets loyalty program members to hold negative point balances for non-qualifying currencies is enabled ( <code>true</code> ) or disabled ( <code>false</code> ) for your org. The default value is <code>false</code> . This field is available in API version 61.0 or later.
<code>enableSegmentQueryByDataGraph</code>	boolean	Indicates whether the capability, which uses Data Cloud data graphs to query for Data Cloud segments that customers are part of, is enabled ( <code>true</code> ) or disabled ( <code>false</code> ). The default value is <code>false</code> . This field is available in API version 62.0 or later.
<code>enablePromSetupProRuleStatusInheritancePolyt</code>	boolean	Indicates whether the capability, which allows Promotion Setup processes and rules to be deployed to target orgs in the same status as the source orgs when using the <code>LoyaltyProgramSetup</code> metadata type, is enabled ( <code>true</code> ) or disabled ( <code>false</code> ). The default value is <code>false</code> . This field is available in API version 62.0 or later.
<code>enableLoyaltyGenerativeAi</code>	boolean	Indicates whether the Einstein AI capabilities of Loyalty Management are enabled ( <code>true</code> ) or not ( <code>false</code> ). The default value is <code>false</code> . This field is available in API version 62.0 or later.

## Declarative Metadata Sample Definition

The following is an example of a IndustriesLoyaltySettings component.

```
<?xml version="1.0" encoding="UTF-8"?>
<IndustriesLoyaltySettings xmlns="http://soap.sforce.com/2006/04/metadata">
  <enableNQPPRealTimePointBalance>true</enableNQPPRealTimePointBalance>
  <enableFixedTypeNQPAggregation>true</enableFixedTypeNQPAggregation>
  <enableQPPRealTimePointBalance>true</enableQPPRealTimePointBalance>

  <enableLoyaltyRulesVerifyCdpMemberSegment>false</enableLoyaltyRulesVerifyCdpMemberSegment>

  <enableLoyaltyRedeemedPointsExpirationInfoPref>true</enableLoyaltyRedeemedPointsExpirationInfoPref>

  <enableLoyaltyServiceExcellence>true</enableLoyaltyServiceExcellence>
  <enableAutomaticVoucherCodeGeneration>true</enableAutomaticVoucherCodeGeneration>

  <enableLoyaltyApiAccessForExternalSiteUsers>false</enableLoyaltyApiAccessForExternalSiteUsers>

  <enableAutomaticMemberTierAssessmentSelection>false</enableAutomaticMemberTierAssessmentSelection>

  <enableNonQualifyingPointsConsolidation>false</enableNonQualifyingPointsConsolidation>

  <enablePointsLifecycleTracking>false</enablePointsLifecycleTracking>
  <enableNegativePointBalance>false</enableNegativePointBalance>
  <enableSegmentQueryByDataGraph>false</enableSegmentQueryByDataGraph>

  <enablePromSetupProcRuleStatusInheritDplymt>false</enablePromSetupProcRuleStatusInheritDplymt>
</IndustriesLoyaltySettings>
```

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>IndustriesLoyalty</members>
    <name>Settings</name>
  </types>
  <version>53.0</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.

## IndustriesUnifiedPromotionsSettings

Represents the settings to Global Promotions Management.

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

In the package manifest, all organization settings metadata types are accessed using the Settings name. See Settings for more details.

## File Suffix and Directory Location

IndustriesUnifiedPromotionsSettings values are stored in a single file named `IndustriesUnifiedPromotions.settings` 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

IndustriesUnifiedPromotionsSettings components are available in API version 60.0 and later.

## Fields

Field Name	Field Type	Description
<code>enableUnifiedPromotions</code>	boolean	Indicates whether Global Promotions Management is enabled ( <code>true</code> ) or disabled ( <code>false</code> ) for your org. The default value is <code>true</code> .
<code>enableGlobalPromotionsProductCatalogManagement</code>	boolean	Indicates whether Product Catalog Management is enabled ( <code>true</code> ) or disabled ( <code>false</code> ) in your org. The default value is <code>false</code> . This field is available in API version 62.0 or later.

## Declarative Metadata Sample Definition

The following is an example of a IndustriesGamificationSettings component.

```
<?xml version="1.0" encoding="UTF-8"?>
<IndustriesUnifiedPromotionsSettings xmlns="http://soap.sforce.com/2006/04/metadata">
  <enableUnifiedPromotions>true</enableUnifiedPromotions>

  <enableGlobalPromotionsProductCatalogManagement>false</enableGlobalPromotionsProductCatalogManagement>
</IndustriesUnifiedPromotionsSettings>
```

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>IndustriesUnifiedPromotions</members>
    <name>Settings</name>
  </types>
  <version>60.0</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 10 Loyalty Management Tooling API Objects

In this chapter ...


- [BenefitAction](#)

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. For more information about Tooling API objects and to find a complete reference of all the supported objects, see [Introducing Tooling API](#).

## BenefitAction

---

Represents details of an action that can be triggered for a benefit. This object is available in API version 51.0 and later.

 **Note:** 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

To use this object, the Loyalty Management license is required.

### Fields

Field	Details
Description	<p><b>Type</b> textarea</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The description of the benefit action.</p>
DeveloperName	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Sort</p> <p><b>Description</b> 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. This field is automatically generated, but you can supply your own value if you create the record using the API.</p> <p>When creating large sets of data, always specify a unique <code>DeveloperName</code> for each record. If no <code>DeveloperName</code> is specified, performance may slow while Salesforce generates one for each record.</p>

Field	Details
FlowDefinition	<p><b>Type</b> picklist</p> <p><b>Properties</b> Filter, Group, Nillable, Restricted picklist, Sort</p> <p><b>Description</b> The ID of the flow associated with the benefit action. The possible values depend on the flows defined in the org.</p>
FlowInstance	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The flow associated with the benefit action.</p>
FullName	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Group, Nillable</p> <p><b>Description</b> The full name of the associated in Metadata API.  Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
IsActive	<p><b>Type</b> boolean</p> <p><b>Properties</b> Defaulted on create, Filter, Group, Sort</p> <p><b>Description</b> Indicates whether the benefit action is active.  The default value is <code>false</code>.</p>
IsBenefitExpirationAllowed	<p><b>Type</b> boolean</p> <p><b>Properties</b> Defaulted on create, Filter, Group, Sort</p> <p><b>Description</b> Indicates whether the benefit action is allowed to expire the benefit.  The default value is <code>false</code>.</p>



Field	Details
IsBenefitUpdateAllowed	<p><b>Type</b> boolean</p> <p><b>Properties</b> Defaulted on create, Filter, Group, Sort</p> <p><b>Description</b> Indicates whether the benefit action is allowed to update the benefit. The default value is <code>false</code>.</p>
Language	<p><b>Type</b> picklist</p> <p><b>Properties</b> Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p><b>Description</b> The language of the benefit action.</p>
ManageableState	<p><b>Type</b> picklist</p> <p><b>Properties</b> Filter, Group, Nillable, Restricted picklist, Sort</p> <p><b>Description</b> Indicates the manageable state of the specified component that is contained in a package. Possible values are:</p> <ul style="list-style-type: none"> <li>• <code>beta</code>—Managed-Beta</li> <li>• <code>deleted</code>—Managed-Proposed-Deleted</li> <li>• <code>deprecated</code>—Managed-Proposed-Deprecated</li> <li>• <code>deprecatedEditable</code>—SecondGen-Installed-Deprecated</li> <li>• <code>installed</code>—Managed-Installed</li> <li>• <code>installedEditable</code>—SecondGen-Installed-Editable</li> <li>• <code>released</code>—Managed-Released</li> <li>• <code>unmanaged</code>—Unmanaged</li> </ul>
MasterLabel	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Sort</p> <p><b>Description</b> The UI label of the benefit action.</p>
MemberBenefitVariableName	<p><b>Type</b> string</p>

Field	Details
	<p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The name of the input variable that stores the member benefit ID in the flow specified for the benefit action. The field is available in API version 52.0 and later.</p>
Metadata	<p><b>Type</b> <a href="#">BenefitAction</a></p> <p><b>Properties</b> Create, Nillable, Update</p> <p><b>Description</b> Provides access to the associated metadata type and related fields in Tooling API.  Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
NamespacePrefix	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> 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 <b><i>namespacePrefix__componentName</i></b> notation.</p>
ProcessType	<p><b>Type</b> picklist</p> <p><b>Properties</b> Defaulted on create, Filter, Group, Restricted picklist, Sort</p> <p><b>Description</b> The process type of the benefit action.  Possible value is:</p> <ul style="list-style-type: none"> <li>• <code>Loyalty</code></li> </ul> <p>The default value is <code>Loyalty</code>.</p>
Type	<p><b>Type</b> picklist</p> <p><b>Properties</b> Defaulted on create, Filter, Group, Restricted picklist, Sort</p>

**Field****Details**

---

**Description**

The type of the benefit action.

Possible value is:

- Flow

The default value is Flow.

---

## CHAPTER 11 Loyalty Management Mobile SDK

### In this chapter ...

- [What to Expect from Loyalty Management Mobile SDK](#)
- [Supported Versions of Tools and Components](#)
- [Supported DateTime Format](#)
- [Getting Started with Loyalty Management Mobile SDK](#)
- [Loyalty Management Mobile SDK for iOS Development](#)
- [Loyalty Management Mobile SDK for Android Development](#)
- [Loyalty Management Sample App for Mobile SDK](#)
- [Receipt Scanning with Loyalty Management Mobile SDK](#)
- [Loyalty Management Mobile SDK and Sample App Support](#)
- [Upgrade Loyalty Management Mobile SDK for iOS](#)

Elevate your customer experience by providing Loyalty Management on iOS and Android mobile devices. Use Loyalty Management Mobile SDK to engage members or business customers with the brand on iOS and Android mobile channels and enhance brand engagement. The standalone is lightweight and provides customer flexibility, immersive mobile user experience, and a ready to use sample app embedded with Loyalty Management features.

Loyalty Management Mobile SDK delivers customized user experiences and provides developers with the tools to build iOS and Android mobile applications. Developers can use the SDK to produce standalone custom apps that they can distribute the app through the App Store or Google Play Store or embed the SDK in an existing app. If you're building an app from scratch, you can integrate the SDK during the development process. If you have an existing app and want to embed the SDK, modify your code to include the functionality of the SDK. To demonstrate the capabilities of Loyalty Management Mobile SDK, a sample app, which is a reference implementation of the SDK is included with the SDK. Developers can refer to these sample apps to understand the usage and best practices while building their own apps. Developers can also reuse the sample app source code that's included within the SDK and extend the code as required. The SDK is designed to work with your existing mobile app infrastructure, and provides a smooth integration process. Whether you build a new app or embed the SDK in an existing app, the SDK helps you create an effective mobile application for your loyalty program.

### SEE ALSO:

[Salesforce Help: Receipt Scanning with Loyalty Management Mobile SDK](#)

[Salesforce Help: Receipt Scanning with Loyalty Management Mobile SDK](#)

### EDITIONS

Available in: all editions that have Loyalty Management enabled

## What to Expect from Loyalty Management Mobile SDK

Loyalty Management Mobile SDK helps a loyalty program to engage with their customers through a mobile app. Use the mobile SDK to manage members' profiles, to track transactions, to reward members, and to manage promotions and vouchers. Manage your loyalty rewards and engage with the brand.

The Loyalty Management Mobile SDK comes along with a built-in MyNTORewards sample app, which is a reference implementation of the SDK. With the sample app, you can:

- Enroll in the loyalty program.
- Review and opt in to active promotional offers, or opt out of a promotion.
- View your membership details, points summary, and tier details.
- View your transaction history.
- View and redeem your active vouchers.
- View the benefits of the loyalty program.
- View badges assigned to you.

### EDITIONS

Available in: all editions that have Loyalty Management enabled

## Supported Versions of Tools and Components

Loyalty Management Mobile SDK works on iOS 15.0+ and Android Studio Electric Eel 2022.1.1+ versions. Install the tools and components required to set up Loyalty Management Mobile SDK

### EDITIONS

Available in: all editions that have Loyalty Management enabled

### iOS

Tool or Component	Supported Version	Installation Details
Swift Version	5.7+	Installed by Xcode
Xcode	14.0+	Install from the App Store
iOS SDK	15.0+	Installed by Xcode
Swift Package Manager	5.7+	Included in Swift

### Android

Tool or Component	Supported Version	Installation Details
Android Studio Electric Eel	2022.1.1+	Install from the Android Developer site
Android SDK and APIs	8.0+ (API level 26+)	Install from the Android Developer site
Android SDK Tools	8.0+	Install from the Android Developer site
Android Virtual Device	8.0+	Install from Android Studio
Kotlin	1.8.0+	Installed by Android Studio

## Supported DateTime Format

---

Use the `yyyy-MM-dd' T' HH:mm:ss.SSSZ` format to specify dateTime fields. Currently, the Loyalty Management Mobile SDK doesn't support defining custom datetime formats.

- `yyyy` is the four-digit year
- `MM` is the two-digit month (01-12)
- `dd` is the two-digit day (01-31)
- `T` is a separator indicating that time-of-day follows
- `HH` is the two-digit hour (00-23)
- `mm` is the two-digit minute (00-59)
- `ss` is the two-digit seconds (00-59)
- `SSS` is the optional three-digit milliseconds (000-999)
- `Z` is the reference UTC timezone

### EDITIONS

Available in: all editions that have Loyalty Management enabled

## Getting Started with Loyalty Management Mobile SDK

---

Prepare to install the Loyalty Management Mobile SDK, to create a connected app, and then to set up your experience cloud users' access to the Loyalty Management Mobile SDK.

### [Prepare to Install Loyalty Management Mobile SDK](#)

Prepare your org before you install Loyalty Management Mobile SDK. Before you install Loyalty Management Mobile SDK, ensure you have a Salesforce org that has Loyalty Management and Experience Cloud enabled.

### [Create a Connected App to Integrate with Loyalty Management](#)

Create a connected app for Loyalty Management Mobile SDK to request access to Salesforce APIs and to enable secure integration with the APIs. For a connected app to request access, integrate the app with the Salesforce API by using the OAuth 2.0 protocol.

### [Prepare Experience Cloud Users for Loyalty Management Mobile SDK](#)

Prepare your Experience Cloud user before you install Loyalty Management Mobile SDK. To register or log in to the sample app for Loyalty Management Mobile SDK, it's recommended that you set up Experience Cloud for your loyalty program member profile.

### EDITIONS

Available in: all editions that have Loyalty Management enabled

## Prepare to Install Loyalty Management Mobile SDK

Prepare your org before you install Loyalty Management Mobile SDK. Before you install Loyalty Management Mobile SDK, ensure you have a Salesforce org that has Loyalty Management and Experience Cloud enabled.

1. Get a Salesforce [Developer Edition organization](#) that has Loyalty Management enabled.
2. To use another program, modify the program name in the Loyalty Management Mobile SDK configuration file.
3. Create an Experience Cloud site user profile for your loyalty program. See [Create Community User Profile](#).
4. [Create a connected app](#) on page 527 and enable an external application to integrate with Salesforce by using APIs and standard protocols.
5. Create and activate loyalty program processes and process rules for GetMemberPromotions, EnrollInPromotion, and OptOutFromPromotion APIs. See [Create Loyalty Program Processes by Using Promotion Setup](#).
6. Access the [Loyalty Management Mobile SDK API References for iOS](#) on page 545 and [Loyalty Management Mobile SDK API References for Android](#) on page 587.
7. Review iOS and Android Native requirements, see [Development Prerequisites for iOS and Android](#).
8. Install the required iOS and Android software. See [iOS Basic Requirements](#) and [Native Android Requirements](#).
9. To set up the Loyalty Management Sample App for Mobile SDK in Android Studio, see [Setting Up Sample Projects in Android Studio](#).

### EDITIONS

Available in: all editions that have Loyalty Management enabled

### USER PERMISSIONS

To enable experience cloud:

- Customize Application

To read, create, update, or delete connected apps:

- Customize Application

AND

Modify All Data

OR

Manage Connected Apps

To create a loyalty program process:

- Loyalty Management

## Create a Connected App to Integrate with Loyalty Management

Create a connected app for Loyalty Management Mobile SDK to request access to Salesforce APIs and to enable secure integration with the APIs. For a connected app to request access, integrate the app with the Salesforce API by using the OAuth 2.0 protocol.

1. From your Salesforce org, in the Quick Find box in Setup, enter *Apps*, and then select **App Manager**.
2. Under Connected Apps, click **New Connected App**.
3. Under Basic Information, enter these values.
  - Connected App Name: Name of your connected app.
  - API Name: Keep the default API name, which is automatically filled based on the connected app name.
  - Contact Email: Your email address.
4. Under API (Enable OAuth Settings), select **Enable OAuth Settings**.
5. In the Callback URL, enter a URL where Salesforce can send OAuth2 responses.
6. Add these scopes to the Selected OAuth Scopes list.
  - Access the identity URL service (id, profile, email, address, phone)
  - Access Visualforce applications (visualforce)

### EDITIONS

Available in: all editions that have Loyalty Management enabled

### USER PERMISSIONS

To read, create, update, or delete connected apps:

- Customize Application

AND

Modify All Data

OR

Manage Connected Apps

- Manage user data via APIs (api)
  - Perform requests at any time (refresh\_token, offline\_access)
7. Select **Enable Authorization Code and Credential Flow**, and deselect **Require Proof Key for Code Exchange (PKCE) Extension for Supported Authorization Flows**, **Require Secret for Web Server Flow**, and **Require Secret for Refresh Token Flow**.
  8. Save the connected app.
  9. From Setup, in the Quick Find box, enter *Connected Apps*, and then select **Manage Connected Apps**.
  10. Click **Edit** next to the connected app that you're configuring access for.
  11. Under OAuth Policies, select **Admin approved users are Pre-authorized** as the permitted users.
  12. Select **Relax IP restrictions** as the IP relaxation.
  13. Select **Refresh token is valid until revoked** as the refresh token policy.
  14. Save your changes.

From the App Manager, for the connected app, click **Manage Consumer Details**, and copy the consumer key and consumer secret. Use these credentials in mobile applications to authenticate with Salesforce APIs.

#### SEE ALSO:

[Mobile SDK Development Guide: Create a Connected App](#)

[Mobile SDK Development Guide: Create a Connected App](#)

## Prepare Experience Cloud Users for Loyalty Management Mobile SDK

Prepare your Experience Cloud user before you install Loyalty Management Mobile SDK. To register or log in to the sample app for Loyalty Management Mobile SDK, it's recommended that you set up Experience Cloud for your loyalty program member profile.

Create a connected app and an Experience Cloud site user for your loyalty program.

1. Enable your Experience Cloud site user to access the connected app.
  - a. From Setup, in the Quick Find box, enter *manage connected*, and then select **Manage Connected Apps**.
  - b. Click the connected app that you created, and then click **Manage Profiles**.
  - c. Select your Experience Cloud site user profile.
  - d. Save your changes.
2. Enable the person account record type to the Experience Cloud site user profile.
  - a. From Setup, click **Users > Profiles**.
  - b. Click your Experience Cloud site user profile.
  - c. In Record Type Settings, find the Accounts entity, and click **Edit**.
  - d. Move the Person Account record type from Available Record Types to Selected Record Types.
  - e. Save your changes.
3. Create Loyalty Program Processes for the GetMemberPromotions, EnrollInPromotion, and OptOutFromPromotion APIs.

#### EDITIONS

Available in: all editions that have Loyalty Management enabled

#### USER PERMISSIONS

To read, create, update, or delete connected apps:

- Customize Application
- AND
- Modify All Data
- OR
- Manage Connected Apps

To create, edit, and delete profiles:

- Manage Profiles
- AND
- Permission Sets



4. Provide the Experience Cloud site user the permission to join or to opt out of promotions from the Experience Cloud site.
  - a. From Setup, in the Quick Find box, enter *Loyalty Management*, and then select **Loyalty Management Settings**.
  - b. Turn on Customers and Members Join and Opt Out from Experience Cloud Site.
5. Enable API access for Experience Cloud users.
  - a. From Setup, click **Users > Profiles**.
  - b. Click your Experience Cloud site user profile.
  - c. Under Administrative Permissions, select **API Enabled**.
  - d. Save your changes.
6. Add permissions for Salesforce Objects.
  - a. From Setup, click **Users > Profiles**.
  - b. Click your Experience Cloud site user profile.
  - c. Under Standard Object Permissions, enable permissions for standard objects.
  - d. Save your changes.
7. Add the sharing rules.
  - a. From Setup, in the Quick Find box, enter *Sharing*, and then select **Sharing Settings**.
  - b. Under Sharing Rules, click **New**.
  - c. Add rules for Journal Type, Loyalty Program, Loyalty Program Process, Promotion, Voucher, Voucher Definition, and Voucher Code.
  - d. Save your changes.

#### [Supported Standard Objects for Experience Cloud Site User Setup](#)

To set up Experience Cloud for your loyalty program member profile, enable permissions for these standard objects.

#### SEE ALSO:

[Salesforce Help: Create a Loyalty Program Process](#)

[Supported Standard Objects for Experience Cloud Site User Setup](#)

[Salesforce Help: Create a Loyalty Program Process](#)

[Supported Standard Objects for Experience Cloud Site User Setup](#)

## Supported Standard Objects for Experience Cloud Site User Setup

To set up Experience Cloud for your loyalty program member profile, enable permissions for these standard objects.

Standard Object	Read	Create	Edit
Accounts	✓	✓	✓
Contacts	✓	✓	✓
Journal SubTypes	✓	✗	✗

#### EDITIONS

Available in: all editions that have Loyalty Management enabled

Standard Object	Read	Create	Edit
Journal Types	✓	✗	✗
Loyalty Ledgers	✓	✗	✗
Loyalty Member Currency	✓	✗	✗
Loyalty Member Tiers	✓	✗	✗
Loyalty Program Currencies	✓	✗	✗
Loyalty Program Processes	✓	✗	✗
Loyalty Tiers	✓	✗	✗
Loyalty Tier Groups	✓	✗	✗
Member Benefits	✓	✗	✗
Promotions	✓	✗	✗
Transaction Journals	✓	✗	✗
Vouchers	✓	✗	✗
Voucher Definitions	✓	✗	✗

## Loyalty Management Mobile SDK for iOS Development

Loyalty Management Mobile SDK for iOS delivers libraries and sample Xcode project to develop mobile apps on iOS. To create the iOS application, Loyalty Management Mobile SDK for iOS uses the ForceSwift library, which uses SwiftUI to build the user interfaces in iOS apps while interacting with Salesforce.

### EDITIONS

Available in: all editions that have Loyalty Management enabled

#### [Clone and Import Loyalty Management Mobile SDK for iOS](#)

Clone the Loyalty Management Mobile SDK GitHub repository for iOS to create a local copy of the repository on your development machine. The repository contains the code for Loyalty Management Mobile SDK for iOS and the MyNTORewards sample app.

#### [Loyalty Management Mobile SDK for iOS Classes](#)

Loyalty Management Mobile SDK for iOS classes organize the functionalities and data structures provided by the SDK. When building applications using the SDK, classes provide components, services, or features that you can interact with.

#### [Loyalty Management Mobile SDK API References for iOS](#)

These API references describe the functions, classes, methods, parameters, and return values of the Loyalty Management Mobile SDK. If your admin has enabled Receipt Make API requests to Salesforce from the POS system to enroll members, view member details, view transactions, manage vouchers, create orders, and accrue and redeem points and vouchers.

## Clone and Import Loyalty Management Mobile SDK for iOS

Clone the Loyalty Management Mobile SDK GitHub repository for iOS to create a local copy of the repository on your development machine. The repository contains the code for Loyalty Management Mobile SDK for iOS and the MyNTORewards sample app.

1. [Clone Mobile SDK for iOS GitHub Repository](#)

Clone the GitHub repository to copy the code base and revision history of the repository to your local computer.

2. [Import SDK into iOS Swift Projects](#)

After you clone the mobile SDK for iOS repository, import Loyalty Management Mobile SDK in Xcode to automatically download and manage external dependencies.

### EDITIONS

Available in: all editions that have Loyalty Management enabled

## Clone Mobile SDK for iOS GitHub Repository

Clone the GitHub repository to copy the code base and revision history of the repository to your local computer.

1. Go to the [Loyalty Management Mobile SDK iOS repository](#).

2. To copy the repository URL to your clipboard, click **Code**, and then click .

3. Open Terminal, and go to the directory where you want to clone the repository.

4. Enter `git clone`

```
https://github.com/salesforce-misc/LoyaltyMobileSDK-iOS
```

5. Press Enter and wait for the repository to clone.

You can now set up your own development environment.

### EDITIONS

Available in: all editions that have Loyalty Management enabled

## Import SDK into iOS Swift Projects

After you clone the mobile SDK for iOS repository, import Loyalty Management Mobile SDK in Xcode to automatically download and manage external dependencies.

1. Open your existing app project in Xcode or create an app project.

2. With your app project open in Xcode, select **File > Swift Packages > Add Package Dependency**.

3. Enter the repository URL:

```
https://github.com/salesforce-misc/LoyaltyMobileSDK-iOS
```

4. Select a version, and click **Add Package** and on the confirmation window, click **Add Package**.

5. To import the LoyaltyMobileSDK module into your Swift file, open the swift file where you want to save LoyaltyMobileSDK, and to the first line of code, add `import LoyaltyMobileSDK`.

6. Press Enter or return.

You can now use the classes and structures in the LoyaltyMobileSDK module and API references to manage the features in your application.

### EDITIONS

Available in: all editions that have Loyalty Management enabled

[Create an App Project in Xcode](#)

If you don't have an existing app project in Xcode, create an app project to import the Loyalty Management Mobile SDK for iOS.

## SEE ALSO:

[Create an App Project in Xcode](#)[Create an App Project in Xcode](#)

## Create an App Project in Xcode

If you don't have an existing app project in Xcode, create an app project to import the Loyalty Management Mobile SDK for iOS.

1. Open Xcode, and click **Create a new Xcode project**.
2. Select **iOS > App**, and then click **Next**.
3. Enter the product name.
4. Select a team.
5. Select **Storyboard** as the interface and **Swift** as the language.
6. Select **Include Tests**, and then click **Next**.

**EDITIONS**

Available in: all editions that have Loyalty Management enabled

## Loyalty Management Mobile SDK for iOS Classes

Loyalty Management Mobile SDK for iOS classes organize the functionalities and data structures provided by the SDK. When building applications using the SDK, classes provide components, services, or features that you can interact with.

[ForceClient](#)

Create authentication network requests by using the ForceClient class. ForceClient uses ForceAuthenticator, which is a protocol that defines the required methods to handle access tokens in Salesforce API.

[Logger](#)

The `Logger` class provides a standardized way to log error, warning, or info messages. Trace the code execution and identify error or debug.

[LoyaltyAPIManager](#)

Manage requests related to loyalty program and interactions between the application and the APIs.

[NetworkManager](#)

The `NetworkManager` class manages network requests and processing of data. The NetworkManager class conforms to the NetworkManagerProtocol, which defines the set of methods and properties that are used to manage network requests and responses.

**EDITIONS**

Available in: all editions that have Loyalty Management enabled

## ForceClient

Create authentication network requests by using the ForceClient class. ForceClient uses ForceAuthenticator, which is a protocol that defines the required methods to handle access tokens in Salesforce API.

Use this signature to define the ForceClient class.

```
public class ForceClient
```

## Define Authentication

Handle the authentication process and encapsulate the logic to obtain and refresh the access token. Use this resource and signature to define the auth variable.

```
auth
```

```
public var auth: ForceAuthenticator
```

## Define the Network Manager

Make HTTP requests to the Salesforce server and manage the responses. Use this resource and signature to define the forceNetworkManager variable.

```
ForceNetworkManager
```

```
public var forceNetworkManager: NetworkManagerProtocol
```

## Create an Instance from Force Authenticator

Create an instance from the ForceAuthenticator, and defines the required methods to handle access tokens in Salesforce API. Use this resource and signature to define the public initializer for the ForceClient class.

```
init(auth:forceNetworkManager:)
```

```
public init(auth: ForceAuthenticator, forceNetworkManager: NetworkManagerProtocol = NetworkManager.shared
```

Use the parameters in this table to define an authentication instance.

Parameters	Description
auth	A <code>ForceAuthenticator</code> instance that defines the required methods to handle access tokens in Salesforce API.
forceNetworkManager	A <code>NetworkManagerProtocol</code> instance that defines the requirements of a network manager.

## Fetch Authentication Requests

Get all the REST requests for authentication. This request returns a decoded JSON response result. Use this resource and signature to define the fetch request.

```
fetch(type:with:urlSession:)
```

```
func fetch<T>(type: T.Type, with request: URLRequest, urlSession: URLSession = .shared)
async throws -> T where T : Decodable
```

Use the parameters in this table to create a fetch request.

Parameters	Description
type	A type or model of data that you want to decode from the JSON format.

Parameters	Description
request	A <code>URLRequest</code> , which is executed by <code>URLSession</code> .

## Logger

The `Logger` class provides a standardized way to log error, warning, or info messages. Trace the code execution and identify error or debug.

Use this signature to define the `Logger` class.

```
public class Logger
```

### Log an Informational Message

Use the message parameter to log an informational message by using this resource and signature.

info

```
public static func info(_ message: String)
```

### Log a Debug Message

Use the message parameter to log a debug message by using this resource and signature.

debug(\_:)

```
public static func debug(_ message: String)
```

### Log an Error Message

Use the message parameter to log an error message by using this resource and signature.

error(\_:)

```
public static func error(_ message: String)
```

### Log a Faulty Message

Use the message parameter to log a faulty message by using this resource and signature.

fault(\_:)

```
public static func fault(_ message: String)
```

## LoyaltyAPIManager

Manage requests related to loyalty program and interactions between the application and the APIs.

Use this signature to define the `LoyaltyAPIManager` class.

```
public class LoyaltyAPIManager
```

## Define Authentication

Create an instance of `ForceAuthenticator` for authentication by using this resource and signature.

`auth`

```
public var auth: ForceAuthenticator
```

## Create a Loyalty Program Name

Create a loyalty program name variable by using this resource and signature.

`loyaltyProgramName`

```
public var loyaltyProgramName: String
```

## Create an Instance URL

Provide the base URL of the loyalty program API by using this resource and signature. Base URL is the root address of your Salesforce instance. For example, if the URL in your browser's address bar is `https://na35.salesforce.com/home/home.jsp`, your base URL is `https://na35.salesforce.com`.

`instanceURL`

```
public var instanceURL: String
```

## Initialize the LoyaltyAPIManager Class

Initialize a `LoyaltyAPIManager` that contains the necessary parameters such as `auth`, `loyaltyProgramName`, `instanceURL`, and `forceClient` by using this resource and signature.

`init(auth:loyaltyProgramName:instanceURL:forceClient:)`

```
public init(auth: ForceAuthenticator, loyaltyProgramName: String, instanceURL: String, forceClient: ForceClient)
```

## Sort Requests

Sort the requests by a specific field by using this resource and signature.

`SortBy`

```
public init(auth: ForceAuthenticator, loyaltyProgramName: String, instanceURL: String, forceClient: ForceClient)
```

Use the parameters in this table to create a sort request.

Parameter	Definition	Description
<code>expirationDate</code>	<code>case expirationDate</code>	Sorts the requests by the expiration date of the authentication token.
<code>effectiveDate</code>	<code>case effectiveDate</code>	Sorts the requests by the effective date of the authentication token.

Parameter	Definition	Description
createdDate	<code>case createdDate</code>	Sorts the requests by the created date of the authentication token.

## Sort Results

Specify if you want to view the sort results in ascending or descending order by using this resource and signature.

```
public enum SortOrder : String
```

Use the parameters in this table to create a sort order request.

Parameter	Definition	Description
ascending	<code>case ascending</code>	Sorts the results in ascending order.
descending	<code>case descending</code>	Sorts the results in descending order.

## Define Voucher Status

Define the possible statuses of a voucher by using this resource and signature.

VoucherStatus

```
public enum VoucherStatus : String
```

Use the parameters in this table to define the voucher statuses.

Parameter	Definition	Description
issued	<code>case issued</code>	Represents the vouchers that are issued.
cancelled	<code>case cancelled</code>	Represents the vouchers that are cancelled.
expired	<code>case expired</code>	Represents the vouchers that have expired.

## Identify Resource Type

Identify the type of resource to request by using this resource and signature. Use these resources to create secure and scalable mobile applications.

Resource

```
public enum Resource
```

Use the methods in this table to get resources for a loyalty program.

Method	Definition	Description
<code>individualEnrollment(programName:version:)</code>	<code>case individualEnrollment(programName: String, version: String)</code>	Creates an individual enrollment for the loyalty program. The <code>individualEnrollment</code> method takes two parameters: <code>programName</code> ,



Method	Definition	Description
		which represents the name of the loyalty program for which the individual enrollment is being created, and <code>version</code> , which represents the API version.
<code>getMemberBenefits (memberId:version:)</code>	<b>case</b> <code>getMemberBenefits (memberId: String, version: String)</code>	Retrieves benefits related to the member ID. The <code>getMemberBenefits</code> method takes two parameters: <code>memberId</code> , which represents the unique identifier of the member whose benefits are being retrieved, and <code>version</code> , which represents the API version.
<code>getMemberProfile (programName:version:)</code>	<b>case</b> <code>getMemberProfile(programName: String, version: String)</code>	Retrieves program name related to the member ID. The <code>getMemberProfile</code> method takes two parameters: <code>programName</code> , which represents the program membership name, and <code>version</code> , which represents the API version.
<code>getTransactionHistory (programName:memberId:version:)</code>	<b>case</b> <code>getTransactionHistory (programName: String, memberId: String, version: String)</code>	Retrieves the transaction history associated with the member ID. The <code>getTransactionHistory</code> method takes three parameters: <code>programName</code> , which represents the loyalty program name, <code>memberId</code> , which represents the unique identifier of the member whose transaction history is being retrieved, and <code>version</code> , which represents the API version.
<code>getPromotions (programName:version:)</code>	<b>case</b> <code>getPromotions (programName: String, version: String)</code>	Retrieves a member's existing promotions. The <code>getPromotions</code> method takes two parameters: <code>programName</code> , which represents the name of the loyalty program, and <code>version</code> , which represents the API version.
<code>enrollInPromotion (programName:version:)</code>	<b>case</b> <code>enrollInPromotion (programName: String, version: String)</code>	Enrolls the member in a promotion. The <code>enrollInPromotion</code> method takes two parameters: <code>programName</code> , which represents the name of the loyalty program, and <code>version</code> , which represents the API version.
<code>unenrollPromotion (programName:version:)</code>	<b>case</b> <code>unenrollPromotion (programName: String, version: String)</code>	Opts out a member from an available promotion. The <code>unenrollInPromotion</code> method takes two parameters: <code>programName</code> , which represents the name of the loyalty

Method	Definition	Description
<code>getVouchers(programName: String, membershipNumber: String, version: String)</code>	<pre>case getVouchers(programName: String, membershipNumber: String, version: String)</pre>	<p>program, and <code>version</code>, which represents the API version.</p> <p>Retrieves vouchers available to the member ID. The <code>getVouchers</code> method takes three parameters: <code>programName</code>, which represents the name of the loyalty program, <code>membershipNumber</code>, which represents the unique identifier of the member whose vouchers are being retrieved, and <code>version</code>, which represents the API version.</p>

## Get API Resource Path

Get the path for the API resource by using this resource and signature. The function uses a resource parameter as input and returns the URL path of the resource as a string.

`getPath(for:)`

```
public func getPath(for resource: Resource) -> String
```

## Get Member Benefits

Retrieve the benefits associated with a loyalty program member for the specified API version, while asynchronously requesting data from the Member Benefits API. Use this resource and signature to get member benefits.

`getMemberBenefits(for:version:devMode:)`

```
public func getMemberBenefits(
    for memberId: String,
    version: String = LoyaltyAPIVersion.defaultVersion,
    devMode: Bool = false) async throws -> [BenefitModel]
```

The `getMemberBenefits` method takes three parameters: `memberId`, which represents the unique identifier of the member whose benefits are being retrieved, `version`, which represents the API version, and the `devMode` parameter, which determines if the request is run in developer mode.

## Get Member Profile

Get the profile details of a loyalty program member, while asynchronously requesting data from the Member Profile API. Use this resource and signature to get member profile details.

`getMemberProfile(for:version:devMode:)`

```
public func getMemberProfile(
    for memberId: String,
    version: String = LoyaltyAPIVersion.defaultVersion,
    devMode: Bool = false) async throws -> ProfileModel
```

The `getMemberProfile` method takes three parameters: `memberId`, which represents the unique identifier of the member whose profile is being retrieved, `version`, which represents the API version, and the `devMode` parameter, which determines if the request is run in developer mode.

## Get Community Member Profile

Retrieve the Experience Cloud user profile associated with a loyalty program member for the specified API version, while asynchronously requesting data from Salesforce. Use this resource and signature to get the Experience Cloud user profile details.

`getCommunityMemberProfile(version:devMode:)`

```
public func getCommunityMemberProfile(
    version: String = LoyaltyAPIVersion.defaultVersion,
    devMode: Bool = false) async throws -> ProfileModel
```

The `getCommunityMemberProfile` method takes two parameters: `version`, which represents the default version of the mobile SDK, and the `devMode` parameter, which determines if the request is run in developer mode.

## Enroll a Member

Enroll a member to a loyalty program, while asynchronously sending data to the Individual Member Enrollments API. Use this resource and signature to enroll a member.

`postEnrollment(membershipNumber:firstName:lastName:email:phone:emailNotification:version:)`

```
public func postEnrollment(
    membershipNumber: String,
    firstName: String,
    lastName: String,
    email: String, phone: String,
    emailNotification: Bool,
    version: String = LoyaltyAPIVersion.defaultVersion) async throws -> EnrollmentOutputModel
```

This table shows the parameters in the method.

Parameter	Description
<code>membershipNumber</code>	Membership number of the member being enrolled.
<code>firstName</code>	First name of the member being enrolled.
<code>lastName</code>	Last name of the member being enrolled.
<code>email</code>	Email address of the member being enrolled.
<code>phone</code>	Phone number of the member being enrolled.
<code>emailNotification</code>	Indicates whether the member wants to receive email notifications.
<code>version</code>	The default API version.

After the enrollment, the request returns the membership number and program member ID.

## Opt In For a Promotion

Enroll a loyalty member to a promotion, while asynchronously sending data to the Enroll for Promotions API. Use this resource and signature to enroll a member.

`enrollIn(promotion:for:version:)`

```
public func enrollIn(
    promotion promotionName: String,
    for membershipNumber: String,
    version: String = LoyaltyAPIVersion.defaultVersion) async throws
```

The `enrollIn` method takes three parameters: `promotionName`, which represents the promotion name the member is being enrolled to, `membershipNumber`, which represents the membership number of the member being enrolled, and `version`, which represents the API version

## Use Promotion ID to Opt Out of a Promotion

Opt out a loyalty member from a promotion by using the promotion ID. Use this resource and signature to opt out of a promotion.

`unenroll(promotionId:for:version:devMode:)`

```
public final func unenroll(
    promotionId: String,
    for membershipNumber: String,
    version: String = LoyaltyAPIVersion.defaultVersion,
    devMode: Bool = false) async throws
```

The `unenroll` method takes two parameters: `promotionID`, which represents the promotion ID from which the member is being unenrolled, `membershipNumber`, which represents the membership number of the member being unenrolled, and `version`, which represents the API version.

## Use Promotion Name to Opt Out of a Promotion

Opt out a loyalty member from a promotion by using the promotion name. Use this resource and signature to opt out of a promotion.

`unenroll(promotionName:for:version:devMode:)`

```
public final func unenroll(
    promotionName: String,
    for membershipNumber: String,
    version: String = LoyaltyAPIVersion.defaultVersion,
    devMode: Bool = false) async throws
```

This table shows the parameters in the `unenroll` method.

Parameter	Description
<code>promotionName</code>	Name of the promotion from which the member is being unenrolled.
<code>membershipNumber</code>	Membership number of the member being unenrolled.
<code>version</code>	Version of the mobile SDK.
<code>devMode</code>	Determines if the request is run in developer mode.

## Use Member ID to Get Enrolled Promotions

Get the promotions that the member is enrolled to, by using the loyalty program member ID while asynchronously requesting data from the Enroll for Promotions API. Use this resource and signature to get the enrolled promotions.

`getPromotions(memberId:version:devMode:)`

```
public func getPromotions(
    memberId: String,
    version: String = LoyaltyAPIVersion.defaultVersion,
    devMode: Bool = false) async throws -> PromotionModel
```

The `getPromotions` method takes three parameters: `memberID`, which represents the unique identifier of the member whose promotions are being retrieved, `version`, which represents the default API version, and `devMode`, which determines if the request is run in developer mode.

## Use Membership Number to Get Enrolled Promotions

Get the promotions that the member is enrolled to, by using the loyalty program membership number while asynchronously requesting data from the Enroll for Promotions API. Use this resource and signature to get the enrolled promotions.

`getPromotions(membershipNumber:version:devMode:)`

```
public func getPromotions(
    membershipNumber: String,
    version: String = LoyaltyAPIVersion.defaultVersion,
    devMode: Bool = false) async throws -> PromotionModel
```

The `getPromotions` method takes three parameters: `membershipNumber`, which represents the membership number of the member whose promotions are being retrieved, `version`, which represents the default API version, and `devMode`, which determines if the request is run in developer mode.

## Get Transaction Journals

Get transaction journals of a loyalty member. Obtain either accrual or redemption transaction journal records. Use this resource and signature to get transactions.

`getTransactions(for:number:journalTypeName:journalSubTypeName:periodStartDate:periodEndDate:version:devMode:)`

```
public func getTransactions(
    for membershipNumber: String,
    pageNumber: Int? = nil,
    journalTypeName: String? = nil,
    journalSubTypeName: String? = nil,
    periodStartDate: String? = nil,
    periodEndDate: String? = nil,
    version: String = LoyaltyAPIVersion.defaultVersion,
    devMode: Bool = false) async throws -> [TransactionJournal]
```

The `getTransactions` method takes the `pageNumber` parameter that represents the number of the page you want returned. If you don't specify a value, the first page is returned. Each page contains 200 transaction journals and the transaction journals are sorted based on the activity date.

## Get Vouchers

Retrieve the vouchers issued to a loyalty member by using this resource and signature.

```
getVouchers(membershipNumber:voucherStatus:pageNumber:productId:productCategoryId:productName:productCategoryName:sortBy:sortOrder:version:devMode:)
```

```
public func getVouchers(
    membershipNumber: String,
    voucherStatus: [VoucherStatus]? = nil,
    pageNumber: Int? = nil,
    productId: [String]? = nil,
    categoryId: [String]? = nil,
    productName: [String]? = nil,
    categoryName: [String]? = nil,
    sortBy: SortBy? = nil,
    sortOrder: SortOrder? = nil,
    version: String = LoyaltyAPIVersion.defaultVersion,
    devMode: Bool = false) async throws -> [VoucherModel]
```

This table shows the parameters in the `getVouchers` method.

Parameter	Description
<code>membershipNumber</code>	Membership number of the member whose vouchers are being retrieved.
<code>voucherStatus</code>	The status of a voucher.
<code>pageNumber</code>	Number of the page you want returned. If you don't specify a value, the first page is returned. By default, each page contains 200 vouchers, which are sorted based on the voucher creation date.
<code>productId</code>	The ID of the product related to the vouchers that you want to retrieve. You can specify the ID of up to 20 products.
<code>categoryId</code>	The ID of the product categories related to the vouchers that you want to retrieve. You can specify the ID of up to 20 product categories.
<code>productName</code>	The product name related to the voucher.
<code>categoryName</code>	The name of the product categories related to the vouchers that you want to retrieve. You can specify the ID of up to 20 product category names.
<code>sortBy</code>	Specifies the field used to sort the vouchers you want to retrieve. The default value is <code>ExpirationDate</code> .
<code>sortOrder</code>	Specifies the sort order of the vouchers you want to retrieve. The default value is <code>Ascending</code> .
<code>version</code>	Version of the API.
<code>devMode</code>	Determines if the request is run in developer mode.

### [Create an Instance of LoyaltyAPIManager Class](#)

Create an instance of `LoyaltyAPIManager` class and interact with the Loyalty Management APIs.

## Create an Instance of LoyaltyAPIManager Class

Create an instance of `LoyaltyAPIManager` class and interact with the Loyalty Management APIs.

1. Create an instance of `LoyaltyAPIManager` by using this signature.

```
let loyaltyAPIManager = LoyaltyAPIManager(auth: forceAuthenticator, loyaltyProgramName:
    "YourLoyaltyProgramName", instanceURL: "YourInstanceURL", forceClient: forceClient)
```

2. Call the appropriate methods to interact with the Loyalty Management API by using this signature.

```
import LoyaltyMobileSDK

let instanceURL = URL(string: "https://your_salesforce_instance_url")!
let loyaltyProgramName = "YourLoyaltyProgramName"
let forceClient = ForceClient(clientId: "your_client_id", clientSecret:
    "your_client_secret", redirectURI: "your_redirect_uri")

let loyaltyAPIManager = LoyaltyAPIManager(instanceURL: instanceURL, loyaltyProgramName:
    loyaltyProgramName, forceClient: forceClient)
```

3. To enroll an individual member, use this signature.

```
let enrollmentOutput = try await loyaltyAPIManager.postEnrollment(
    membershipNumber: membershipNumber,
    firstName: firstName,
    lastName: lastName,
    email: email,
    phone: phone,
    emailNotification: emailNotification)
```

For example:

```
let membershipNumber = "1234567890"
let firstName = "John"
let lastName = "Doe"
let email = "john.doe@example.com"
let phone = "4157891234"
let emailNotification = true
```

4. To retrieve the member benefits, use this signature.

```
let benefits = try await loyaltyAPIManager.getMemberBenefits(for: "memberId")
```

5. To retrieve the member profile, use this signature.

```
let profile = try await loyaltyAPIManager.getMemberProfile(for: "memberId")
```

6. To retrieve the Experience Cloud user profile, use this signature.

```
let communityProfile = try await loyaltyAPIManager.getCommunityMemberProfile()
```

7. To opt a member in to a promotion, use this signature.

```
try await loyaltyAPIManager.enrollIn(promotion: "PromotionName", for: "1234567890")
```

8. To opt out a member from a promotion using promotion ID or promotion name, use this signature.

```
try await loyaltyAPIManager.unenroll(promotionId: "promotionId", for: "1234567890")
try await loyaltyAPIManager.unenroll(promotionName: "PromotionName", for: "1234567890")
```

9. To retrieve loyalty member transactions, use this signature.

```
let transactions = try await loyaltyAPIManager.getTransactions(for: "1234567890")
```

10. To retrieve promotions for a loyalty member, use this signature.

```
let promotions = try await loyaltyAPIManager.getPromotions(membershipNumber: "1234567890")
```

11. To retrieve vouchers for a loyalty member, use this signature.

```
let vouchers = try await loyaltyAPIManager.getVouchers(membershipNumber: "1234567890")
```

## NetworkManager

The `NetworkManager` class manages network requests and processing of data. The `NetworkManager` class conforms to the `NetworkManagerProtocol`, which defines the set of methods and properties that are used to manage network requests and responses.

Use this signature to define the `NetworkManager` class.

```
public class NetworkManager : NetworkManagerProtocol
```

### Create a Shared Instance

Create a shared instance of the `NetworkManager` class by using this signature.

shared

```
public static let shared: NetworkManager
```

### Get REST Requests

Use `Async/Await` to get all the REST requests asynchronously by using this signature.

fetch (type:request:urlSession:)

```
public func fetch<T>(type: T.Type, request: URLRequest, urlSession: URLSession =
URLSession.shared) async throws -> T where T : Decodable
```

Use the parameters in this table to create a fetch request.

Parameters	Description
type	A type or model of data that you want to decode from the JSON format.
request	The <code>URLRequest</code> , which is executed by <code>URLSession</code> .



Parameters	Description
urlSession	An instance of <code>URLSession</code> , which is used to execute the request and manage HTTP requests and responses. The default instance is <code>URLSession.shared</code> .

## Loyalty Management Mobile SDK API References for iOS

These API references describe the functions, classes, methods, parameters, and return values of the Loyalty Management Mobile SDK. If your admin has enabled Receipt Make API requests to Salesforce from the POS system to enroll members, view member details, view transactions, manage vouchers, create orders, and accrue and redeem points and vouchers.

### EDITIONS

Available in: all editions that have Loyalty Management enabled.

#### [Individual Member Enrollments](#)

Enroll an individual member in a loyalty program, while asynchronously sending data to the Individual Member Enrollments API.

#### [Enroll for Promotions](#)

Enroll loyalty program member to a promotion, while asynchronously sending data to the Enroll for Promotions API.

#### [Expense Analysis](#)

Stores the product and information from a receipt submitted for review by a member. To get the receipt information, access the `LoyaltyProgramMember` object reference.

#### [Get Member Promotions](#)

Retrieve a list of promotions that the member is eligible for, while asynchronously requesting data from the Get Member Promotions loyalty program process API.

#### [Member Benefits](#)

Retrieve a list of benefits for a member, while asynchronously requesting data from the Member Benefits API.

#### [Member Profile](#)

Get the profile details of a loyalty program member, while asynchronously requesting data from the Member Profile API.

#### [Member Vouchers](#)

Retrieve a list of vouchers issued to the member, while asynchronously requesting data from the Member Vouchers API.

#### [Receipt Status Updation](#)

Update the status of the receipt, reward points, and create a transaction journal for each line item in the receipt.

#### [Receipt Upload](#)

Upload and scan a receipt with a receipt scanning app that uses optical character recognition to read and interpret receipts. For the unlocked package, this API uploads the image into the Amazon S3 bucket.

#### [Transaction History](#)

Retrieve member's transaction history, while asynchronously requesting data from the Transaction History API.

#### [Opt Out from a Promotion](#)

Opt out a loyalty program member from an available promotion.

## Individual Member Enrollments

Enroll an individual member in a loyalty program, while asynchronously sending data to the Individual Member Enrollments API.

## Resource

```
postEnrollment(membershipNumber:firstName:lastName:email:phone:emailNotification:version:)
```

## Signature

```
public func postEnrollment(
    membershipNumber: String,
    firstName: String,
    lastName: String,
    email: String,
    phone: String,
    emailNotification: Bool,
    version: String = LoyaltyAPIVersion.defaultVersion) async throws -> EnrollmentOutputModel
```

## Input Parameters

Parameter	Type	Description	Required or Optional
email	String	Email address of the member being enrolled.	Required
emailNotification	Boolean	Indicates whether the member wants to receive email notifications ((true)) or not ((false)).	Required
firstName	String	First name of the member being enrolled.	Required
lastName	String	Last name of the member being enrolled.	Required
membershipNumber	String	Membership number of the member being enrolled.	Required
phone	String	Phone number of the member being enrolled.	Required
version	String	Version of the API. The request body is available in version 55.0 and later.	Optional

## Return Value

An EnrollmentOutputModel instance of type:

```
public struct EnrollmentOutputModel : Codable
```

## Sample Output

```
{
  "contactId": "0034x00001JdPg6",
  "loyaltyProgramMemberId": "01M4x000000LOrM",
  "loyaltyProgramName": "NTO Insider",
  "membershipNumber": "Member2",
  "transactionJournals": [
    {
      "activityDate": "2022-01-01T08:00:00.000Z",
      "journalSubType": "Member Enrollment",
      "journalType": "Accrual",
      "loyaltyProgram": "NTO Insider",
      "loyaltyProgramMember": "Member2",
      "referredMember": null,
      "status": "Pending",
      "transactionJournalId": "01V4x000000CcBF"
    }
  ]
}
```

## Output Parameters

Parameter	Type	Description
activityDate	String	The date and time of when transaction journal is created.
contactId	String	The ID of the loyalty program member's person account.
journalSubType	String	The journal subtype of the transaction journal.
journalType	String	The journal type of the transaction journal.
loyaltyProgram	String	The loyalty program related with the transaction journal.
loyaltyProgramMember	String	The membership number of the loyalty program related to the transaction journal.
loyaltyProgramMemberId	String	The ID of the loyalty program member record created for the member.
loyaltyProgramName	String	The name of the loyalty program into which the member is enrolled.
membershipNumber	String	The membership number of the enrolled member.
referredMember	String	The membership number of the loyalty program member who has been referred by the member.

Parameter	Type	Description
status	String	The status of the transaction journal.
transactionJournalId	String	The ID of the transaction journal that is created.

## Enroll for Promotions

Enroll loyalty program member to a promotion, while asynchronously sending data to the Enroll for Promotions API.

### Resource

```
enrollIn(promotion:for:version:)
```

### Signature

```
public func enrollIn(
    promotionName: String,
    membershipNumber: String,
    version: String = LoyaltyAPIVersion.defaultVersion) async throws
```

### Input Parameters

Parameter	Type	Description	Required or Optional
promotionName	String	Name of the promotion that the member wants to enroll into.	Required
membershipNumber	String	Unique identifier of the member.	Required
version	String	Version of the API. The request body is available in version 55.0 and later.	Optional

### Sample Output

```
{
  "message": null,
  "outputParameters": {
    "outputParameters": {
      "results": [
        {
          "MemberId": "11p4x00000002ffAAA"
        }
      ]
    }
  },
  "simulationDetails": {},
  "status": true
}
```

```
}

```

## Output Parameter

Parameter	Type	Description
memberId	String	The ID of the member enrolled for the promotion.

## Expense Analysis

Stores the product and information from a receipt submitted for review by a member. To get the receipt information, access the LoyaltyProgramMember object reference.

## Resource

```
services/apexrest/expense-analysis
```

## Rest Method Type

POST

## Signature

```
curl --location --request POST
'https://yourInstance.salesforce.com/services/apexrest/expense-analysis/?filename=8925687d-7489-7eb-5307-19d4b33c650.jpeg&membershipnumber=124345696'
\
--header 'Authorization: Bearer
00DB000000FX0aR!ARQAQCyXQGgMcxjFvjshnPCDJEZyPavtzxM26XvjDU.gr56iTgkGp_EQCZ0yyi1QZzILLDq0lmy3KpJqhKAOCc56o4jqx6CxU'
\
--header 'Cookie: BrowserId=MZ4W4WebEe6bX0_gtfh0Tw; CookieConsentPolicy=0:1;
LSKey-c$CookieConsentPolicy=0:1'
```

## Request Parameters

Parameter	Type	Description	Required or Optional
filename	String	The file that contains the image of the receipt.	Required
membershipnumber	String	The unique identifier of the member.	Required

## Sample Output

```
{
  "storeName" : "alpine nutrition",
  "storeAddress" : "625 Market Street,\n6th floor,\nSan Francisco,\nCA, 94105",
```

```

"receiptSFDCId" : "a0PHp00000tstqeMAA",
"receiptNumber" : "#0529-0002",
"receiptDate" : "2020-01-10",
"memberShipNumber" : "124345696",
"lineItem" : [ {
  "quantity" : "5",
  "productName" : "Alpine-Bran Cereal",
  "price" : "$7.99",
  "lineItemPrice" : "$39.95",
  "isEligible" : false
}, {
  "quantity" : "2",
  "productName" : "Alpine-Oat Cereal",
  "price" : "$7.99",
  "lineItemPrice" : "$15.98",
  "isEligible" : false
} ],
"confidenceStatus" : "Partial"
}

```

## Output Parameters

Parameter	Type	Description
confidenceStatus	String	The receipt analysis status based on the confidence value defined in the app.  Possible values are: <ul style="list-style-type: none"> <li>• Success</li> <li>• Failure</li> <li>• Partial</li> </ul>
isEligible	Boolean	Indicates whether the item is eligible for accrual of loyalty points ( <code>true</code> ) or not ( <code>false</code> ).  This value is displayed only if the admin has opted to display ineligible items in the receipt.
lineItem	String	The details of the receipt line item.
lineItemPrice	String	The total price of the receipt line item.
memberShipNumber	String	The unique identifier of the member.
price	Currency	The unit price of the receipt line item.
productName	String	The name of the product purchased.
quantity	Integer	The quantity of the product purchased.
receiptDate	dateTime	The date when the receipt is generated.

Parameter	Type	Description
receiptNumber	String	The receipt number.
receiptSFDCId	Reference	The unique ID of the receipt.
storeAddress	String	The store address.
storeName	String	The store name.

**SEE ALSO:**

[Salesforce Help: Set Up Amazon Web Services](#)

[Salesforce Help: Set Up Salesforce Org With Loyalty Management Receipt Scanner Kit Unlocked Package](#)

[Salesforce Help: Set Up Amazon Web Services](#)

[Salesforce Help: Set Up Salesforce Org With Loyalty Management Receipt Scanner Kit Unlocked Package](#)

## Get Member Promotions

Retrieve a list of promotions that the member is eligible for, while asynchronously requesting data from the Get Member Promotions loyalty program process API.

### Resource

Here's the request when the input parameter is member ID.

```
getPromotions(memberId:version:devMode:)
```

### Signature

```
public func getPromotions(
    memberId: String,
    version: String = LoyaltyAPIVersion.defaultVersion,
    devMode: Bool = false) async throws -> PromotionModel
```

Here's the request when the input parameter is membership number.

```
getPromotions(memberNumber:version:devMode:)
```

### Signature

```
public func getPromotions(
    membershipNumber: String,
    version: String = LoyaltyAPIVersion.defaultVersion,
    devMode: Bool = false) async throws -> PromotionModel
```

## Input Parameters

Parameter	Types	Description	Required or Optional
devMode	Boolean	Indicates whether the request is run in developer mode ((true)) or not ((false)).	Optional
memberId	String	Identifier of the member whose promotions are being retrieved.	Required
membershipNumber	String	The membership number of the member whose promotions are being retrieved.	Required
version	String	Version of the API. The request body is available in version 55.0 and later.	Optional

## Sample Output

```
{
  "message": null,
  "outputParameters": {
    "outputParameters": {
      "results": [
        {
          "loyaltyPromotionType": "STANDARD",
          "maximumPromotionRewardValue": 0,
          "totalPromotionRewardPointsVal": 0,
          "loyaltyProgramCurrency": "01c4x000000L1bmAAC",
          "memberEligibilityCategory": "Eligible",
          "promotionEnrollmentRqr": true,
          "fulfillmentAction": "CREDIT_POINTS",
          "promotionName": "Anniversary promotion",
          "promotionId": "0c84x000000CoNhAAK",
          "startDate": "2020-01-01",
          "endDate": "2022-12-31",
          "description": "Double point promotion on member purchases on their anniversary",
          "imageName": "offer2"
        },
        {
          "loyaltyPromotionType": "CUMULATIVE",
          "maximumPromotionRewardValue": 0,
          "totalPromotionRewardPointsVal": 200,
          "loyaltyProgramCurrency": "01c4x000000L1bmAAC",
          "memberEligibilityCategory": "Eligible",
          "promotionEnrollmentRqr": false,
          "fulfillmentAction": "CREDIT_POINTS",
          "promotionName": "Referral Promotions",
          "promotionId": "0c84x000000CoNiAAK",
          "startDate": "2021-01-02",

```



```

        "description": "Additional points for member referral",
        "imageName": "promotion1"
    },
    {
        "promEnrollmentStartDate": "2022-11-01",
        "memberEligibilityCategory": "EligibleButNotEnrolled",
        "promotionEnrollmentEndDate": "2022-11-30",
        "promotionEnrollmentRqr": true,
        "promotionName": "New Year Promotion Premium",
        "promotionId": "0c84x000000CtzRAAS",
        "startDate": "2022-11-01",
        "description": "New Year Promotion Premium",
        "imageName": "meall1"
    },
    {
        "promEnrollmentStartDate": "2022-11-01",
        "memberEligibilityCategory": "Eligible",
        "promotionEnrollmentEndDate": "2022-11-30",
        "promotionEnrollmentRqr": true,
        "promotionName": "Christmas Promotion",
        "promotionId": "0c84x000020CtzWAAS",
        "startDate": "2022-12-01",
        "description": "Christmas Promotion - Happy holidays, enjoy a joyful
holiday season with this promotion",
        "imageName": "promotion3"
    },
    {
        "loyaltyPromotionType": "STANDARD",
        "maximumPromotionRewardValue": 0,
        "totalPromotionRewardPointsVal": 0,
        "loyaltyProgramCurrency": "01c4x000000L1bmAAC",
        "memberEligibilityCategory": "Eligible",
        "promotionEnrollmentRqr": true,
        "fulfillmentAction": "CREDIT_POINTS",
        "promotionName": "Anniversary promotion",
        "promotionId": "0c84x004000CoNhAAK",
        "startDate": "2020-01-01",
        "description": "Double point promotion on member purchases on their
anniversary",
        "imageName": "offers"
    },
    {
        "loyaltyPromotionType": "CUMULATIVE",
        "maximumPromotionRewardValue": 0,
        "totalPromotionRewardPointsVal": 200,
        "loyaltyProgramCurrency": "01c4x000000L1bmAAC",
        "memberEligibilityCategory": "Eligible",
        "promotionEnrollmentRqr": false,
        "fulfillmentAction": "CREDIT_POINTS",
        "promotionName": "Referral Promotions",
        "promotionId": "0c84x000006CoNiAAK",
        "startDate": "2021-01-02",
        "description": "Additional points for member referral",
        "imageName": "shoes1"
    }

```

```

    },
    {
      "promEnrollmentStartDate": "2022-11-01",
      "memberEligibilityCategory": "EligibleButNotEnrolled",
      "promotionEnrollmentEndDate": "2022-11-30",
      "promotionEnrollmentRqr": true,
      "promotionName": "Thanksgiving Promotion Premium",
      "promotionId": "0c84x070000CtzRAAS",
      "startDate": "2022-11-01",
      "description": "Thanksgiving Promotion Premium",
      "imageName": "jean1"
    },
    {
      "promEnrollmentStartDate": "2022-11-01",
      "memberEligibilityCategory": "Ineligible",
      "promotionEnrollmentEndDate": "2022-11-30",
      "promotionEnrollmentRqr": true,
      "promotionName": "New Year Promotion Premium",
      "promotionId": "0c84x000000CtzWAAS",
      "startDate": "2022-11-01",
      "description": "New Year Promotion Premium - Ineligible",
      "imageName": "coffeel"
    }
  ]
}
},
"simulationDetails": {},
"status": true
}

```

## Output Parameters

Parameter	Type	Description
description	String	The description of the promotion.
endDate	String	The end date of the promotion.
fulfillmentAction	String	Specifies the fulfillment action of the related promotion. Possible values are: Credit_Points and Issue_Voucher
imageName	String	The name for the promotion image.
loyaltyProgramCurrency	String	The loyalty program currency associated with the promotion.
loyaltyPromotionType	String	The type of promotion. Possible values are: <ul style="list-style-type: none"> <li>Cumulative</li> <li>Standard</li> <li>Joint</li> </ul>

Parameter	Type	Description
maximumPromotionRewardValue	Double	The value that can be accumulated for the promotion.
memberEligibilityCategory	String	Specifies whether the loyalty program member is eligible for the promotion. Possible values are: <ul style="list-style-type: none"> <li>• Eligible</li> <li>• EligibleButNotEnrolled</li> <li>• Ineligible</li> </ul>
message	String	The request response message.
promEnrollmentStartDate	String	The date when the enrollment in the related promotion starts.
promotionEnrollmentEndDate	String	The date when the enrollment in the related promotion ends.
promotionEnrollmentRqr	Boolean	Indicates whether the related promotion requires member enrollment.
promotionId	String	The ID related to the promotion.
promotionImageURL	String	The URL for the promotion image. In the promotion record, enter an image URL and ensure that the webpage that hosts the image is publicly accessible.
promotionName	String	The name of the promotion.
startDate	String	The start date of the related promotion.
status	Boolean	Indicates whether the member is eligible to enroll for the promotion.
totalPromotionRewardPointsVal	Double	The total points credited to the loyalty program member for the related promotion.

## Member Benefits

Retrieve a list of benefits for a member, while asynchronously requesting data from the Member Benefits API.

### Resource

```
getMemberBenefits (for:version:devMode:)
```

## Signature

```
public func getMemberBenefits(
    for memberId: String,
    version: String = LoyaltyAPIVersion.defaultVersion,
    devMode: Bool = false) async throws -> [BenefitModel]
```

## Input Parameters

Parameter	Type	Description	Required or Optional
devMode	Boolean	Indicates whether the request is run in developer mode (true) or not (false).	Optional
memberId	String	Identifier of the member whose benefits are being retrieved.	Required
version	String	Version of the API. The request body is available in version 55.0 and later.	Optional

## Return Value

A BenefitModel instance of type:

```
public struct BenefitModel : Identifiable, Codable
```

## Sample Output

```
{
  "memberBenefits": [
    {
      "benefitId": "0ji4x0000008REdAAM",
      "benefitName": "Extended Returns (7 Days)",
      "benefitTypeId": "0jh4x0000008REQAA2",
      "benefitTypeName": "Extended Returns",
      "createdRecordId": null,
      "createdRecordName": null,
      "endDate": null,
      "isActive": true,
      "memberBenefitStatus": null,
      "startDate": null
    },
    {
      "benefitId": "0ji4x0000008REjAAM",
      "benefitName": "Standard Support",
      "benefitTypeId": "0jh4x0000008REXAA2",
      "benefitTypeName": "Customer Support",
      "createdRecordId": null,
      "createdRecordName": null,
      "endDate": null,
    }
  ]
}
```

```

        "isActive": true,
        "memberBenefitStatus": null,
        "startDate": null
    },
    {
        "benefitId": "0ji4x0000008REeAAM",
        "benefitName": "Free Subscription",
        "benefitTypeId": "0jh4x0000008REXAA2",
        "benefitTypeName": "Free Subscription",
        "createdRecordId": null,
        "createdRecordName": null,
        "endDate": null,
        "isActive": true,
        "memberBenefitStatus": null,
        "startDate": null
    },
    {
        "benefitId": "0ji4x0000008REZAA2",
        "benefitName": "Free Shipping",
        "benefitTypeId": "0jh4x0000008REQAA2",
        "benefitTypeName": "Free Shipping",
        "createdRecordId": null,
        "createdRecordName": null,
        "endDate": null,
        "isActive": true,
        "memberBenefitStatus": null,
        "startDate": null
    },
    {
        "benefitId": "0ji4x0000008REiAAM",
        "benefitName": "Tier Exclusive Offer",
        "benefitTypeId": "0jh4x0000008REXAA2",
        "benefitTypeName": "Tier Exclusive Offer",
        "createdRecordId": null,
        "createdRecordName": null,
        "endDate": null,
        "isActive": true,
        "memberBenefitStatus": null,
        "startDate": null
    },
    {
        "benefitId": "0ji4x0000008REfAAM",
        "benefitName": "Complimentary Voouchers",
        "benefitTypeId": "0jh4x0000008REQAA2",
        "benefitTypeName": "Complimentary Voouchers",
        "createdRecordId": null,
        "createdRecordName": null,
        "endDate": null,
        "isActive": true,
        "memberBenefitStatus": null,
        "startDate": null
    }
]
}

```

## Output Parameters

Parameter	Type	Description
benefitId	String	The ID of the benefit available for the member.
benefitName	String	The name of the benefit available for the member.
benefitTypeId	String	The ID of the associated benefit type record.
benefitTypeName	String	The name of the associated benefit type.
createdRecordId	String	The ID of the record created during benefit assignment.
createdRecordName	String	The display name of the record created during benefit assignment.
description	String	The description of the benefit available to a loyalty program member.
endDate	String	The date the benefit expires.
isActive	Boolean	Indicates whether the benefit is active ( <code>true</code> ) or not ( <code>false</code> ).
memberBenefitStatus	String	The status of the benefit available for the member.
startDate	String	The date from when the benefit applies to a member.

## Member Profile

Get the profile details of a loyalty program member, while asynchronously requesting data from the Member Profile API.

### Resource

```
getMemberProfile(for:version:devMode:)
```

### Signature

```
public func getMemberProfile(
    for memberId: String,
    version: String = LoyaltyAPIVersion.defaultVersion,
    devMode: Bool = false) async throws -> ProfileModel
```

## Input Parameters

Parameter	Type	Description	Required or Optional
devMode	Boolean	Indicates whether the request is run in developer mode (true) or not (false).	Optional
memberId	String	Identifier of the member whose benefits are being retrieved.	Required
version	String	Version of the API. The request body is available in version 55.0 and later.	Optional

## Return Value

A ProfileModel instance of type:

```
public struct ProfileModel : Codable
```

## Sample Output

```
{
  "additionalLoyaltyProgramMemberFields": {
    "City__c": null,
    "Age__c": null,
    "Anniversary__c": null,
    "ReferredMemberPromotion__c": null,
    "DateOfBirth__c": null,
    "Hobbies__c": null,
    "Gender__c": null,
    "SeedData__c": false,
    "Income__c": null,
    "State__c": null,
    "B2CRetail_Most_Recent_Survey_Email_Date__c": null
  },
  "associatedAccount": null,
  "associatedContact": {
    "contactId": "0034x00001EhiXSAAZ",
    "email": "ab@gmail.com",
    "firstName": "Aman",
    "lastName": "Bindal"
  },
  "canReceivePartnerPromotions": true,
  "canReceivePromotions": true,
  "enrollmentChannel": "Email",
  "enrollmentDate": "2022-01-01",
  "groupCreatedByMember": null,
  "groupName": null,
  "lastActivityDate": null,
  "loyaltyProgramMemberId": "01M4x000000LECAEA4",
  "loyaltyProgramName": "NTO Insider",
```

```

"memberCurrencies": [
  {
    "additionalLoyaltyMemberCurrencyFields": {},
    "escrowPointsBalance": 0.0,
    "expirablePoints": 0.0,
    "lastAccrualProcessedDate": null,
    "lastEscrowProcessedDate": null,
    "lastExpirationProcessRunDate": null,
    "lastPointsAggregationDate": null,
    "lastPointsResetDate": null,
    "loyaltyMemberCurrencyName": "Reward Points",
    "loyaltyProgramCurrencyId": "01c4x000000L1bmAAC",
    "loyaltyProgramCurrencyName": null,
    "memberCurrencyId": "01z4x000000LC3nAAG",
    "nextQualifyingPointsResetDate": null,
    "pointsBalance": 17850,
    "qualifyingPointsBalanceBeforeReset": 0.0,
    "totalEscrowPointsAccrued": 0.0,
    "totalEscrowRolloverPoints": 0.0,
    "totalPointsAccrued": 0.0,
    "totalPointsExpired": 0.0,
    "totalPointsRedeemed": 0.0
  },
  {
    "additionalLoyaltyMemberCurrencyFields": {},
    "escrowPointsBalance": 0.0,
    "expirablePoints": 0.0,
    "lastAccrualProcessedDate": null,
    "lastEscrowProcessedDate": null,
    "lastExpirationProcessRunDate": null,
    "lastPointsAggregationDate": null,
    "lastPointsResetDate": null,
    "loyaltyMemberCurrencyName": "Tier Points",
    "loyaltyProgramCurrencyId": "01c4x000000L1bnAAC",
    "loyaltyProgramCurrencyName": null,
    "memberCurrencyId": "01z4x000000LC3oAAG",
    "nextQualifyingPointsResetDate": "2023-01-01",
    "pointsBalance": 4000,
    "qualifyingPointsBalanceBeforeReset": 0.0,
    "totalEscrowPointsAccrued": 0.0,
    "totalEscrowRolloverPoints": 0.0,
    "totalPointsAccrued": 0.0,
    "totalPointsExpired": 0.0,
    "totalPointsRedeemed": 0.0
  }
],
"memberStatus": "Active",
"memberTiers": [
  {
    "additionalLoyaltyMemberTierFields": {},
    "areTierBenefitsAssigned": false,
    "loyaltyMemberTierId": "01y4x000000btVzAAI",
    "loyaltyMemberTierName": "Silver",
    "tierChangeReason": null,

```



```

        "tierChangeReasonType": null,
        "tierEffectiveDate": "2022-01-01",
        "tierExpirationDate": "2023-08-31",
        "tierGroupId": "01t4x000000L1bXAAS",
        "tierGroupName": null,
        "tierId": "01g4x000000L1bmAAC",
        "tierSequenceNumber": 10
    },
    ],
    "memberType": "Individual",
    "membershipEndDate": "2023-01-01",
    "membershipLastRenewalDate": null,
    "membershipNumber": "Member1",
    "referredBy": null,
    "relatedCorporateMembershipNumber": null,
    "transactionJournalStatementFrequency": "Monthly",
    "transactionJournalStatementLastGeneratedDate": null,
    "transactionJournalStatementMethod": "Mail"
}

```

## Output Parameters

Property Name	Type	Description
additionalLoyaltyProgramMemberFields	Map<String, Object>	The details of the custom fields added in the loyalty program member object.
associatedAccount	Codable	The details of the account record that is associated with the loyalty program member.
associatedContact	Codable	The details of the contact record that is associated with the loyalty program member.
canReceivePartnerPromotions	Boolean	Indicates whether the loyalty program member can receive partner promotions.
canReceivePromotions	Boolean	Indicates whether the loyalty program member can receive promotions ( <code>true</code> ) or not ( <code>false</code> ).
enrollmentChannel	String	The channel used by the member to enroll in the loyalty program.
enrollmentDate	String	The date on which the member enrolled in the loyalty program.
groupCreatedByMember	String	The membership number of the loyalty program member who created the group.
groupName	String	The name of the loyalty program group to which the member belongs.
lastActivityDate	String	The date on which the member performed their last activity.
loyaltyProgramMemberId	String	The ID of the loyalty program member.
loyaltyProgramName	String	The name of the loyalty program.

Property Name	Type	Description
memberCurrencies	Codable	The currencies associated with the loyalty program member.
memberStatus	String	The status of the loyalty program member.
memberTiers	Codable	The tiers associated with the loyalty program member.
memberType	String	The type of loyalty program member.
membershipEndDate	String	The date on which the membership ends.
membershipLastRenewalDate	String	The date on which the loyalty program member last renewed their membership.
membershipNumber	String	The membership number of the loyalty program member.
referredBy	String	The membership number of the loyalty program member who referred the member to the loyalty program.
relatedCorporateMembershipNumber	String	The membership number of the Corporate type loyalty program member associate with the member.
transactionJournalStatementFrequency	String	The frequency at which transaction journal statements must be delivered to the member.  Possible values are: <ul style="list-style-type: none"> <li>• Monthly</li> <li>• Quarterly</li> </ul>
transactionJournalStatementLastGeneratedDate	String	The date on which the member's last transaction journal statement was generated.
transactionJournalStatementMethod	String	The method used to deliver transaction journal statements to the member.

## Member Vouchers

Retrieve a list of vouchers issued to the member, while asynchronously requesting data from the Member Vouchers API.

### Resource

```
getVouchers(membershipNumber:voucherStatus:pageNumber:productId:productCategoryId:productName:productCategoryName:sortBy:sortOrder:version:obMode:)
```

### Signature

```
public func getVouchers(
    membershipNumber: String,
    voucherStatus: [VoucherStatus]? = nil,
    pageNumber: Int? = nil,
```

```

productId: [String]? = nil,
productCategoryId: [String]? = nil,
productName: [String]? = nil,
productCategoryName: [String]? = nil,
sortBy: SortBy? = nil,
sortOrder: SortOrder? = nil,
version: String = LoyaltyAPIVersion.defaultVersion,
devMode: Bool = false) async throws -> [VoucherModel]

```

## Input Parameters

Parameter Name	Type	Description	Required or Optional
membershipNumber	String	Unique identifier of the member whose vouchers are being retrieved.	Required
devMode	Boolean	Indicates whether the request is run in developer mode ( <code>true</code> ) or not ( <code>false</code> ).	Optional
pageNumber	Integer	The page number to be returned. If you don't specify a value, the first page is returned. Each page contains 200 vouchers, sorted by their date of creation.	Optional
productCategoryId	String	The ID of a product category associated with the member vouchers to be retrieved. You can specify up to 20 product category IDs.	Optional
productCategoryName	String	The product category name associated with the vouchers to be retrieved. You can specify up to 20 product categories.	Optional
productId	String	The product ID associated with the vouchers to be retrieved. You can specify up to 20 product IDs.	Optional
productName	String	The product name associated with the vouchers to be retrieved. You can specify up to 20 product names.	Optional
voucherStatus	String	The status of the member vouchers to filter the API response. Possible values are: <ul style="list-style-type: none"> <li>• Issued</li> <li>• Cancelled</li> <li>• Expired</li> </ul>	Optional
sortBy	String	The voucher field for sorting the API response. The default value is <code>ExpirationDate</code> . Possible values are: <ul style="list-style-type: none"> <li>• CreatedDate</li> <li>• EffectiveDate</li> <li>• ExpirationDate</li> </ul>	Optional

Parameter Name	Type	Description	Required or Optional
sortOrder	String	The sort order of the API response. The default value is Ascending. Possible values are: <ul style="list-style-type: none"> <li>Ascending</li> <li>Descending</li> </ul>	Optional
version	String	Version of the API. The request body is available in version 55.0 and later.	Optional

## Sample Output

```
{
  "voucherCount": 4,
  "vouchers": [
    {
      "description": "20% discount on children group party at selected event centers.",
      "discountPercent": 50,
      "effectiveDate": "2023-01-01",
      "expirationDate": "2023-04-01",
      "isVoucherDefinitionActive": true,
      "isVoucherPartiallyRedeemable": false,
      "promotionId": "0c8RO0000000ILdYAM",
      "promotionName": "Birthday Promotion",
      "redeemedValue": 0.0,
      "remainingValue": 0.0,
      "status": "Issued",
      "type": "DiscountPercentage",
      "voucherCode": "VOU10101011",
      "voucherDefinition": "Birthday Discount Voucher",
      "voucherId": "0kDRO00000000Hk2AI",
      "voucherImageUrl":
"https://thumbs.dreamstime.com/b/10-percent-discount-24471598.jpg",
      "voucherNumber": "00000001"
    },
    {
      "description": "20% discount on children group party at selected event centers.",
      "discountPercent": 40,
      "effectiveDate": "2023-01-01",
      "expirationDate": "2023-04-01",
      "isVoucherDefinitionActive": true,
      "isVoucherPartiallyRedeemable": false,
      "promotionId": "0c8RO0000000ILdYAM",
      "promotionName": "Birthday Promotion",
      "redeemedValue": 0.0,
      "remainingValue": 0.0,
      "status": "Redeemed",
      "type": "DiscountPercentage",
      "voucherCode": "VOU10101012",
      "voucherDefinition": "Birthday Discount Voucher",

```

```

        "voucherId": "0kDRO00000000Hp2AI",
        "voucherImageUrl":
"https://thumbs.dreamstime.com/b/10-percent-discount-24471598.jpg",
        "voucherNumber": "00000002"
    },
    {
        "description": "20% discount on children group party at selected event centers.",

        "discountPercent": 40,
        "effectiveDate": "2023-01-01",
        "expirationDate": "2023-06-01",
        "isVoucherDefinitionActive": true,
        "isVoucherPartiallyRedeemable": false,
        "promotionId": "0c8RO00000000ILdYAM",
        "promotionName": "Birthday Promotion",
        "redeemedValue": 0.0,
        "remainingValue": 0.0,
        "status": "Redeemed",
        "type": "DiscountPercentage",
        "voucherCode": "VOU10101012",
        "voucherDefinition": "Birthday Discount Voucher",
        "voucherId": "0kDRO00000000Hp2BI",
        "voucherImageUrl":
"https://thumbs.dreamstime.com/b/10-percent-discount-24471598.jpg",
        "voucherNumber": "00000003"
    },
    {
        "description": "20% discount on children group party at selected event centers.",

        "discountPercent": 40,
        "effectiveDate": "2023-01-01",
        "expirationDate": "2023-05-01",
        "isVoucherDefinitionActive": true,
        "isVoucherPartiallyRedeemable": false,
        "promotionId": "0c8RO00000000ILdYAH",
        "promotionName": "Birthday Promotion",
        "redeemedValue": 0.0,
        "remainingValue": 0.0,
        "status": "Expired",
        "type": "DiscountPercentage",
        "voucherCode": "VOU10101012",
        "voucherDefinition": "Birthday Discount Voucher",
        "voucherId": "0kD43872679RO000000000Hp2AI",
        "voucherImageUrl":
"https://thumbs.dreamstime.com/b/10-percent-discount-24471598.jpg",
        "voucherNumber": "00000004"
    }
]
}

```

## Output Parameters

Property Name	Type	Description
attributesUrl	String	The URL of the attribute associated with the voucher.
currencyIsoCode	String	The currency ISO code in which the member has redeemed the voucher.
description	String	The description of the voucher issued to the member.
discountPercent	Integer	The discount percentage that's offered to the member by the voucher.
effectiveDate	String	The date the member can redeem the voucher.
expirationDate	String	The date the voucher expires.
faceValue	Double	The value that the member can redeem using the voucher.
id	String	The ID of the voucher record.
isVoucherDefinitionActive	Boolean	Indicates whether the voucher definition associated with the voucher is active ( <code>true</code> ) or not ( <code>false</code> ).
isVoucherPartiallyRedeemable	Boolean	Indicates whether the voucher can be redeemed across multiple transactions ( <code>true</code> ) or only one transaction ( <code>false</code> ).
partnerAccount	String	The name of the partner associated with the voucher's voucher definition.
product	String	The name of the product associated with the voucher definition.
productCategory	String	The product category associated with the voucher's voucher definition.
productCategoryId	String	The ID of product category associated with the voucher's voucher definition.
productId	String	The ID of the product associated with the voucher definition.
promotionId	String	The ID of the promotion for which the voucher is issued.
promotionName	String	The name of the promotion for which the voucher is issued.
redeemedValue	Double	The value that the member has already redeemed from a partially redeemable voucher.
remainingValue	Double	The remaining value of the voucher after partial redemption.
status	String	The status of the voucher.
type	String	The type of voucher issued to the member.
useDate	String	The date the voucher was redeemed.
voucherCode	String	The code of the voucher.

Property Name	Type	Description
voucherDefinition	String	The name of the voucher definition used to issue the voucher to the member.
voucherImageUrl	String	The URL of the image associated with the voucher definition. In the promotion record, enter an image URL and ensure that the webpage that hosts the image is publicly accessible.
voucherNumber	String	The unique number of the voucher.

## Receipt Status Updation

Update the status of the receipt, reward points, and create a transaction journal for each line item in the receipt.

### Resource

```
services/apexrest/receipt-status-updation
```

### HTTP methods

PUT

### Signature

```
curl --location --request PUT
'https://yourInstance.salesforce.com/services/apexrest/receipt-status-updation/' \
--header 'Content-Type: text/plain' \
--header 'Authorization: Bearer
00DB000000FX0aR!ARQACCyXCGMcxjFvjslhnPCDEZyPavtzxM26XvjDU.gr56iTgkGp_EQCZ0yyi1QZzILDqOLmy3KpJqhKAOc56o4jqx6CxU'
\
--header 'Cookie: BrowserId=MZ4W4WebEe6bX0_gtfh0Tw; CookieConsentPolicy=0:1;
LSKey-c$CookieConsentPolicy=0:1' \
--data '{
  "receiptId": "a0N1Q00000d0Rv4UAE",
  "status": "In Progress",
  "comments": ""
}'
```

### Input Parameters

Parameter	Type	Description	Required or Optional
comments	String	The comments related to the receipt status change.	Required
receiptId	String	The ID of the receipt whose status is to be updated.	Required

Parameter	Type	Description	Required or Optional
status	String	The status that's assigned to the receipt after the update. Valid status values are: <ul style="list-style-type: none"> <li>• Approved</li> <li>• Cancelled</li> <li>• Draft</li> <li>• In Progress</li> <li>• Manual Review</li> <li>• Pending</li> <li>• Processed</li> <li>• Rejected</li> </ul>	Required

## Sample Output

```
{
  "storeName" : "alpine nutrition",
  "storeAddress" : "625 Market Street,\n6th floor,\nSan Francisco,\nCA, 94105",
  "receiptSFDCId" : "a0PHp00000tstqeMAA",
  "receiptNumber" : "#0529-0002",
  "receiptDate" : "2020-01-10",
  "memberShipNumber" : "124345696",
  "lineItem" : [ {
    "quantity" : "5",
    "productName" : "Alpine-Bran Cereal",
    "price" : "$7.99",
    "lineItemPrice" : "$39.95",
    "isEligible" : false
  }, {
    "quantity" : "2",
    "productName" : "Alpine-Oat Cereal",
    "price" : "$7.99",
    "lineItemPrice" : "$15.98",
    "isEligible" : false
  } ],
  "confidenceStatus" : "Partial"
}
```

## Output Parameter

Parameter	Type	Description
confidenceStatus	String	The receipt analysis status based on the confidence value defined in the app. Possible values are: Success, Failure, or Partial.



Parameter	Type	Description
<code>isEligible</code>	Boolean	Indicates whether the item is eligible for accrual of loyalty points ( <code>true</code> ) or not ( <code>false</code> ).  This value is displayed only if the admin has opted to display ineligible items in the receipt.
<code>lineItem</code>	String	The details of the receipt line item.
<code>lineItemPrice</code>	String	The total price of the receipt line item.
<code>memberShipNumber</code>	String	The unique identifier of the member.
<code>price</code>	Currency	The unit price of the receipt line item.
<code>productName</code>	String	The name of the product purchased.
<code>quantity</code>	Integer	The quantity of the product purchased.
<code>receiptDate</code>	dateTime	The date when the receipt is generated.
<code>receiptNumber</code>	String	The receipt number.
<code>receiptSFDCId</code>	Reference	The unique ID of the receipt.
<code>storeAddress</code>	String	The store address.
<code>storeName</code>	String	The store name.

**SEE ALSO:**

[Salesforce Help: Set Up Amazon Web Services](#)

[Salesforce Help: Set Up Salesforce Org With Loyalty Management Receipt Scanner Kit Unlocked Package](#)

[Salesforce Help: Set Up Amazon Web Services](#)

[Salesforce Help: Set Up Salesforce Org With Loyalty Management Receipt Scanner Kit Unlocked Package](#)

## Receipt Upload

Upload and scan a receipt with a receipt scanning app that uses optical character recognition to read and interpret receipts. For the unlocked package, this API uploads the image into the Amazon S3 bucket.

### Resource

```
services/apexrest/upload-receipt
```

### Rest Method Type

PUT

## Signature

```
curl --location --request PUT
'https://yourInstance.salesforce.com/services/apexrest/upload-receipt/?membershipnumber=124345696'
\
--header 'Content-Type: image/jpeg' \
--header 'Authorization: Bearer
00DB00000FX0aR!ARQACyXQGgMxjFvjslhnPCDEZyPavtzxM26XvjDU.gr56iTgkGp_EQCZ0yyi1QZzILLDqOlmy3KpJqhKAOc56o4jqx6CxU'
\
--header 'Cookie: BrowserId=MZ4W4WebEe6bX0_gtfh0Tw; CookieConsentPolicy=0:1;
LSKey-c$CookieConsentPolicy=0:1' \
--data '@1jFvSuLyR/Alpine receipt (4) (1) (2).jpg'
```

## Input Parameters

Parameter	Type	Description	Required or Optional
membershipnumber	String	The unique identifier of the member.	Required

## Sample Output

```
{
  "status" : "Success",
  "message" : "b9746ab0-fea3-b9ef-dc9c-ea5b905a822e.jpeg",
  "errorCode" : ""
}
```

## Output Parameter

Parameter	Type	Description
errorCode	Integer	The error code if the transaction fails for any reason.
message	String	The error message if the transaction fails for any reason.
status	String	The status of the receipt upload.

### SEE ALSO:

[Salesforce Help: Set Up Amazon Web Services](#)

[Salesforce Help: Set Up Salesforce Org With Loyalty Management Receipt Scanner Kit Unlocked Package](#)

[Salesforce Help: Set Up Amazon Web Services](#)

[Salesforce Help: Set Up Salesforce Org With Loyalty Management Receipt Scanner Kit Unlocked Package](#)

## Transaction History

Retrieve member's transaction history, while asynchronously requesting data from the Transaction History API.

## Resource

```
getTransactions(for:number:journalTypeName:journalSubTypeName:periodStartDate:periodEndDate:version:devMode:)
```

## Signature

```
public func getTransactions(
    for membershipNumber: String,
    pageNumber: Int? = nil,
    journalTypeName: String? = nil,
    journalSubTypeName: String? = nil,
    periodStartDate: String? = nil,
    periodEndDate: String? = nil,
    version: String = LoyaltyAPIVersion.defaultVersion,
    devMode: Bool = false) async throws -> [TransactionJournal]
```

## Input Parameters

Parameter Name	Type	Description	Required or Optional
devMode	Boolean	Indicates whether the request is run in developer mode ( <code>true</code> ) or not ( <code>false</code> ).	Optional
journalSubTypeName	String	The subtype of the transaction journal record.	Optional
journalTypeName	String	The type of the transaction journal record. Possible values are: <ul style="list-style-type: none"> <li>• <code>Accrual</code></li> <li>• <code>Redemption</code></li> </ul>	Optional
membershipNumber	String	Unique identifier of the member whose transactions are being retrieved.	Required
pageNumber	Integer	The page number to be returned. If you don't specify a value, the first page is returned. Each page contains 200 transactions, sorted by their date of creation.	Optional
periodStartDate	String	Retrieve transaction journals starting on this date.	Optional
periodEndDate	String	Retrieve transaction journals until this date.	Optional
version	String	Version of the API. The request body is available in version 55.0 and later.	Optional

## Sample Output

```
{
  "message": null,
  "status": true,
  "transactionJournalCount": 4,
  "transactionJournals": [
```

```

    {
      "activityDate": "2023-04-08T04:59:40.000Z",
      "additionalTransactionJournalAttributes": [],
      "journalTypeName": "Manual Points Adjustment",
      "pointsChange": [
        {
          "changeInPoints": 100.0,
          "loyaltyMemberCurrency": "Reward Points"
        }
      ],
      "transactionJournalId": "01VRO00000002og2AA",
      "transactionJournalNumber": "00000035"
    },
    {
      "activityDate": "2023-04-08T04:58:07.000Z",
      "additionalTransactionJournalAttributes": [],
      "journalTypeName": "Manual Points Adjustment",
      "pointsChange": [
        {
          "changeInPoints": 300.0,
          "loyaltyMemberCurrency": "Reward Points"
        }
      ],
      "transactionJournalId": "01VRO00000002ob2AA",
      "transactionJournalNumber": "00000034"
    },
    {
      "activityDate": "2023-04-08T04:56:38.000Z",
      "additionalTransactionJournalAttributes": [],
      "journalTypeName": "Manual Points Adjustment",
      "pointsChange": [
        {
          "changeInPoints": -100.0,
          "loyaltyMemberCurrency": "Reward Points"
        }
      ],
      "transactionJournalId": "01VRO00000002oW2AQ",
      "transactionJournalNumber": "00000033"
    },
    {
      "activityDate": "2022-03-08T04:53:19.000Z",
      "additionalTransactionJournalAttributes": [],
      "journalTypeName": "Manual Points Adjustment",
      "pointsChange": [
        {
          "changeInPoints": 100.0,
          "loyaltyMemberCurrency": "Reward Points"
        }
      ],
      "transactionJournalId": "01VRO00000002oR2AQ",
      "transactionJournalNumber": "00000032"
    }
  ]
}

```

## Output Parameters

Parameter Name	Type	Description
externalTransactionNumber	String	The transaction journal record ID from an external source system.
message	String	The message explaining why the request failed.
status	Boolean	Indicates whether the request succeeded ( <code>true</code> ) or failed ( <code>false</code> ).
transactionJournalCount	Integer	The total number of transaction journals that match the input criteria.
transactionJournals	Map<String, Object>[]	The details of the transaction journals.

## Opt Out from a Promotion

Opt out a loyalty program member from an available promotion.

### Resource to Opt Out of a Promotion by Using the Promotion ID

```
unenroll(promotionId:for:version:devMode:)
```

### Signature

```
public final func unenroll(
    promotionId: String,
    for membershipNumber: String,
    version: String = LoyaltyAPIVersion.defaultVersion,
    devMode: Bool = false) async throws
```

### Resource to Opt Out of a Promotion by Using the Promotion Name

```
unenroll(promotionName:for:version:devMode:)
```

### Signature

```
public final func unenroll(
    promotionName: String,
    for membershipNumber: String,
    version: String = LoyaltyAPIVersion.defaultVersion,
    devMode: Bool = false) async throws
```

## Input Parameters

Parameter	Type	Description	Required or Optional
<code>devMode</code>	Boolean	Indicates whether the request is run in developer mode ( <code>true</code> ) or not ( <code>false</code> ).	Optional
<code>membershipNumber</code>	String	Unique identifier of the member who wants to opt out of a promotion	Required
<code>promotionId</code>	String	ID of the promotion that the member wants to opt out of.	Required
<code>promotionName</code>	String	Name of the promotion that the member wants to opt out of.	Required
<code>version</code>	String	Version of the API. The request body is available in version 55.0 and later.	Optional

## Sample Output

```
{
  "message" : null,
  "outputParameters" : {
    "outputParameters" : {
      "results" : [ {
        "LoyaltyProgramMbrPromotionId" : "1lpT10000000h4IAA"
      } ]
    }
  },
  "simulationDetails" : { },
  "status" : true
}
```

## Output Parameter

Parameter	Type	Description
<code>LoyaltyProgramMbrPromotionId</code>	String	The ID of the promotion that the member opts out of.

# Loyalty Management Mobile SDK for Android Development

Loyalty Management Mobile SDK for Android delivers source code, build scripts, and native sample app to develop mobile apps on Android.

To upgrade the Loyalty Management Mobile SDK for Android, see [Update your tools with the SDK Manager](#).

### EDITIONS

Available in: all editions that have Loyalty Management enabled

### [Clone and Import Loyalty Management Mobile SDK for Android](#)

Clone the Loyalty Management Mobile SDK GitHub repository for Android to create a local copy of the repository on your development machine. The repository contains the code for the Loyalty Management Mobile SDK for Android and the MyNTORewards sample app. To create the Android application, Loyalty Management Mobile SDK for Android uses Android Studio to build the user interfaces in Android apps while interacting with Salesforce.

### [Loyalty Management Mobile SDK for Android Classes](#)

The Android classes organize the Loyalty Management Mobile SDK's functionalities and data structures. When developing mobile apps using the SDK, these classes provide components, services, or features with which you can interact.

### [Loyalty Management Mobile SDK API References for Android](#)

The Loyalty Management Mobile SDK for Android interacts with Salesforce Loyalty Management APIs and retrieves member benefits, transactions, profiles, promotions, vouchers, enrolls an individual member, opts a member out of a promotion.

## Clone and Import Loyalty Management Mobile SDK for Android

Clone the Loyalty Management Mobile SDK GitHub repository for Android to create a local copy of the repository on your development machine. The repository contains the code for the Loyalty Management Mobile SDK for Android and the MyNTORewards sample app. To create the Android application, Loyalty Management Mobile SDK for Android uses Android Studio to build the user interfaces in Android apps while interacting with Salesforce.

### EDITIONS

Available in: all editions that have Loyalty Management enabled

### [Clone Mobile SDK for Android GitHub Repository](#)


Clone the GitHub repository to copy the code base and revision history of the repository to your local computer.

### [Import SDK into Android Studio](#)

After you clone the mobile SDK for Android repository, import Loyalty Management Mobile SDK in Android Studio to automatically download and manage external dependencies.

## Clone Mobile SDK for Android GitHub Repository

Clone the GitHub repository to copy the code base and revision history of the repository to your local computer.

1. Go to the [Loyalty Management Mobile SDK Android repository](#).
2. To copy the repository URL to your clipboard, click **Code**, and then click .
3. Open Terminal, and go to the directory where you want to clone the repository.
4. Enter `git clone https://github.com/salesforce-misc/LoyaltyMobileSDK-Android`
5. Press Enter and wait for the repository to clone.

You can now set up your own development environment.

### EDITIONS


Available in: all editions that have Loyalty Management enabled

## Import SDK into Android Studio

After you clone the mobile SDK for Android repository, import Loyalty Management Mobile SDK in Android Studio to automatically download and manage external dependencies.

1. Unzip the `LoyaltyMobileSDK-Android` package that you downloaded.
2. Open Android Studio.
3. Go to **File > New > Import Project**, and then select the `LoyaltyMobileSDK-Android` folder.
4. Click **Open**.
5. Open the `SampleApps/MyNTORewards/build.gradle` file and under Dependencies block, to the first line of Dependencies, add:
 

```
{implementation fileTree(dir: 'libs', include: ['*.jar']) implementation project(path: ':Sources') }
```
6. Open the `SampleApps/MyNTORewards/connectedapp.properties.sample` file and modify the properties of the connected app as per your org.
7. Rename the `connectedapp.properties.sample` file to `connectedapp.properties`.
8. Click **Sync Now**.  
Android Studio automatically downloads and manages external dependencies for the Mobile SDK and the Sample App.
9. Open the SDK files and ensure that the files include the appropriate SDK packages.

 **Note:** If the sync fails, disconnect the VPN to ensure there's no blocker when downloading the dependencies, and then resync.

10. Add the code snippet to add the appropriate SDK package:

```
import com.salesforce.loyalty.mobile.sources.forceUtils.* import
com.salesforce.loyalty.mobile.sources.loyaltyAPI.* import
com.salesforce.loyalty.mobile.sources.loyaltyModels.* import
com.salesforce.loyalty.mobile.sources.loyaltyExtensions.*
```

You can now use the classes and structures in the `LoyaltyMobileSDK-Android` module and API references to manage the features in your application.

## Loyalty Management Mobile SDK for Android Classes

The Android classes organize the Loyalty Management Mobile SDK's functionalities and data structures. When developing mobile apps using the SDK, these classes provide components, services, or features with which you can interact.

### Logger

The `Logger` class provides a standardized way to save debug, verbose, info, or error log messages.

### LoyaltyClient

Create authentication network requests by using the `LoyaltyClient` class. `LoyaltyClient` uses `ForceAuthenticator`, which is an interface that defines the methods to handle access tokens in the Salesforce API. `LoyaltyClient` creates a retrofit instance to invoke Force APIs.

### EDITIONS

Available in: all editions that have Loyalty Management enabled

### EDITIONS

Available in: all editions that have Loyalty Management enabled



[LoyaltyAPIManager](#)

Manage requests related to loyalty program and interactions between the application and the APIs. The `LoyaltyAPIManager` class manages requests related to loyalty programs using the `FORCE` API and manages authentication by using the instance of `ForceAuthenticator` that was injected at the time of initialization. Use this class to manage asynchronous requests by using Retrofit and Kotlin coroutines.

[LoyaltyAPIInterface](#)

The `LoyaltyApiInterface` class defines the interface for creating, formatting, and sending REST requests to the Salesforce service. This class stores REST API call definitions for all Loyalty API.

## Logger

The `Logger` class provides a standardized way to save debug, verbose, info, or error log messages.

Use this signature to define the `Logger` class.

```
object Logger
```

### Log a Debug Message

Use the `message` and `tag` parameters to log an informational message by using this code.

```
final Unit d(String tag, String message)
```

### Log an Error Message

Use the `message` parameter to log an error message by using this code.

```
final Unit e(String tag, String message, Throwable throwable)
```

### Log an Information Message

Use the `message` parameter to add an information log by using this code.

```
final Unit i(String tag, String message)
```

### Add a Verbose Log

Use the `message` parameter to log a verbose message by using this code.

```
final Unit v(String tag, String message)
```

## LoyaltyClient

Create authentication network requests by using the `LoyaltyClient` class. `LoyaltyClient` uses `ForceAuthenticator`, which is an interface that defines the methods to handle access tokens in the Salesforce API. `LoyaltyClient` creates a retrofit instance to invoke Force APIs.

Use this signature to define the `LoyaltyClient` class.

```
public final class LoyaltyClient
extends NetworkClient
```

#### EDITIONS

Available in: all editions that have Loyalty Management enabled

## getNetworkClient Method

Gives an implementation of the `retrofit` interface using an instance of `retrofit`.

```
LoyaltyApiInterface getNetworkClient()
```

## LoyaltyAPIManager

Manage requests related to loyalty program and interactions between the application and the APIs. The `LoyaltyAPIManager` class manages requests related to loyalty programs using the `Force` API and manages authentication by using the instance of `ForceAuthenticator` that was injected at the time of initialization. Use this class to manage asynchronous requests by using Retrofit and Kotlin coroutines.

Use this signature to define the `LoyaltyAPIManager` class.

```
public final class LoyaltyAPIManager
```

## Method Summary

Modifier and Type	Definition	Description
<code>final Result&lt;EnrollmentResponse&gt;</code>	<code>enroll(String membershipNumber, String promotionName)</code>	Creates an individual enrollment for the loyalty program.
<code>final Result&lt;MemberBenefitsResponse&gt;</code>	<code>getMemberBenefits(String membershipNumber)</code>	Retrieves benefits related to the member ID.
<code>final Result&lt;MemberProfileResponse&gt;</code>	<code>getMemberProfile(String membershipNumber)</code>	Retrieves profile details related to the member ID.
<code>final Result&lt;TransactionsResponse&gt;</code>	<code>getMemberTransactions(String membershipNumber)</code>	Retrieves transactions related to the member. Makes an asynchronous request for transaction history from Salesforce APIs.
<code>final Result&lt;PromotionsResponse&gt;</code>	<code>getEligiblePromotions(String membershipNumber)</code>	Retrieves promotions that a promotion is eligible to enroll for.
<code>final Result&lt;EnrollPromotionsResponse&gt;</code>	<code>enrollPromotions(String membershipNumber, String promotionName)</code>	Enrolls member in a promotion.
<code>final Result&lt;UnenrollPromotionResponse&gt;</code>	<code>unenrollPromotion(String membershipNumber, String promotionName)</code>	Opts out a member from an available promotion.
<code>final Result&lt;VoucherResult&gt;</code>	<code>getAvailableVouchers(String membershipNumber)</code>	Retrieves vouchers are available to the program member.

## enrollInPromotions Method

Enroll a loyalty member to a promotion, while asynchronously sending data to the Enroll for Promotions API. Use this code to enroll a member.

```
final Result<EnrollPromotionsResponse> enrollInPromotions(String membershipNumber, String promotionName)
```

Parameter	Type	Description	Required or Optional
membershipNumber	String	The membership number of the loyalty program member.	Required
promotionName	String	The promotion name the member is being enrolled to	

### getEligiblePromotions Method

Get promotions that a member is eligible to enroll for. Use this code to get eligible promotions.

```
final Result<PromotionsResponse> getEligiblePromotions(String membershipNumber, String memberId)
```

Parameter	Type	Description	Required or Optional
memberId	Boolean	The identifier of the member whose eligible promotions are being retrieved.	Optional
membershipNumber	String	The membership number of the loyalty program member.	Required

### getMemberBenefits Method

Retrieve the benefits associated with a loyalty program member for the specified API version, while asynchronously requesting data from the Member Benefits API. Use this code to get member benefits.

```
final Result<MemberBenefitsResponse> getMemberBenefits(String memberId, String membershipNumber)
```

Parameter	Type	Description	Required or Optional
memberId	Boolean	Identifier of the member whose benefits are being retrieved.	Optional
membershipNumber	String	The membership number of the loyalty program member.	Required

### getMemberProfile Method

Get the profile details of a loyalty program member by using this code.

```
final Result<MemberProfileResponse> getMemberProfile(String memberId, String membershipNumber, String programCurrencyName)
```

Parameter	Type	Description	Required or Optional
memberId	Boolean	Identifier of the member whose profile is being retrieved.	Optional

Parameter	Type	Description	Required or Optional
membershipNumber	String	The membership number of the loyalty program member.	Required
programCurrencyName	String	The name of the loyalty program currency associated with the member.	Optional

## getTransactions Method

Get transaction journals of a loyalty member. Obtain either accrual or redemption transaction journal records. Use this code to get transactions.

```
final Result<MemberBenefitsResponse> getMemberBenefits(String memberId, String membershipNumber)
```

Parameter	Type	Description	Required or Optional
journalSubTypeName	String	The subtype of the transaction journal record.	Optional
journalTypeName	String	The type of the transaction journal record. Possible values are: <ul style="list-style-type: none"> <li>• Accrual</li> <li>• Redemption</li> </ul>	Optional
pageNumber	Integer	The page number to be returned. If you don't specify a value, the first page is returned. Each page contains 200 transactions, sorted by their date of creation.	Optional
membershipNumber	String	The membership number of the loyalty program member.	Required
periodStartDate	String	Retrieve transaction journals starting on this date.	Optional
periodEndDate	String	Retrieve transaction journals until this date.	Optional

## getVouchers Method

Retrieve the vouchers issued to a loyalty member by using this code.

```
final Result<VoucherResult> getVouchers(String membershipNumber, Array<String> voucherStatus, Integer pageNumber, Array<String> productId, Array<String> productCategoryId, Array<String> productName, Array<String> productCategoryName)
```

This table shows the parameters in the `getVouchers` method.

Parameter	Description
<code>membershipNumber</code>	Membership number of the member whose vouchers are being retrieved.
<code>voucherStatus</code>	The status of a voucher.
<code>pageNumber</code>	Number of the page you want returned. If you don't specify a value, the first page is returned. By default, each page contains 200 vouchers, which are sorted based on the voucher creation date.
<code>productId</code>	The ID of the product related to the vouchers that you want to retrieve. You can specify the ID of up to 20 products.
<code>productCategoryId</code>	The ID of the product categories related to the vouchers that you want to retrieve. You can specify the ID of up to 20 product categories.
<code>productName</code>	The product name related to the voucher.
<code>productCategoryName</code>	The name of the product categories related to the vouchers that you want to retrieve. You can specify the ID of up to 20 product category names.

## postEnrollment Method

Enroll a member to a loyalty program, while asynchronously sending data to the Individual Member Enrollments API. Use this code to enroll a member.

```
final Result<EnrollmentResponse> postEnrollment(String firstName, String lastName, String
    email, Map<String, Object> additionalContactAttributes, Boolean emailNotification,
    MemberStatus memberStatus, Boolean createTransactionJournals,
    TransactionalJournalStatementFrequency transactionalJournalStatementFrequency,
    TransactionalJournalStatementMethod transactionalJournalStatementMethod, EnrollmentChannel
    enrollmentChannel, Boolean canReceivePromotions, Boolean canReceivePartnerPromotions)
```

This table shows the parameters in the `postEnrollment` method.

Parameter	Description
<code>additionalContactAttributes</code>	The person account record details of the individual type member.
<code>canReceivePartnerPromotions</code>	Indicates whether the member can receive partner promotions for the program ( <code>true</code> ) or not ( <code>false</code> ).
<code>canReceivePromotions</code>	Indicates whether the loyalty program member can receive promotions.
<code>createTransactionJournals</code>	Indicates whether to create the transaction journal records <code>true</code> ) or not ( <code>false</code> ).
<code>email</code>	The email address of the member being enrolled.
<code>emailNotification</code>	Indicates whether the member wants to receive email notifications.

Parameter	Description
enrollmentChannel	The channel used to by loyalty program members to enroll in to the loyalty program. Possible values are: <ul style="list-style-type: none"> <li>• Pos</li> <li>• Web</li> <li>• Email</li> <li>• CallCenter</li> <li>• Social</li> <li>• Mobile</li> <li>• Store</li> <li>• Franchise</li> <li>• Partner</li> <li>• Print</li> </ul>
firstName	The first name of the member being enrolled.
lastName	The last name of the member being enrolled.
memberStatus	The status of the loyalty program member. Possible values are: <ul style="list-style-type: none"> <li>• Active</li> <li>• Inactive</li> <li>• Custom</li> </ul>
transactionalJournalStatementFrequency	The frequency at which transaction journal statements must be delivered to the member. Possible values are: <ul style="list-style-type: none"> <li>• Monthly</li> <li>• Quarterly</li> </ul>
transactionalJournalStatementMethod	The method used to deliver transaction journal statements to the member. Possible values are: <ul style="list-style-type: none"> <li>• Mail</li> <li>• Email</li> </ul>

### unEnrollPromotion Method

Opt out a loyalty member from a promotion. Use this code to opt out of a promotion.

```
final Result<UnenrollPromotionResponse> unEnrollPromotion(String membershipNumber, String
promotionName)
```

Parameter	Type	Description	Required or Optional
membershipNumber	String	Unique identifier of the member who wants to opt out of a promotion	Required
promotionName	String	Name of the promotion that the member wants to opt out of.	Required

### [Create an Instance of LoyaltyAPIManager Class for Android](#)

Create an instance of the `LoyaltyAPIManager` class and interact with the Loyalty Management APIs.

## Create an Instance of LoyaltyAPIManager Class for Android

Create an instance of the `LoyaltyAPIManager` class and interact with the Loyalty Management APIs.

1. Create an instance of `LoyaltyAPIManager` by using this signature.

```
val loyaltyAPIManager = LoyaltyAPIManager(
    auth = forceAuthManager,
    instanceUrl = mInstanceUrl,
    loyaltyClient = LoyaltyClient(forceAuthManager, mInstanceUrl)
)````
```

2. To interact with the Loyalty Management API, call the appropriate methods by using this signature.

```
import LoyaltyMobileSDK

val forceAuthManager = ForceAuthManager(applicationContext)
val mInstanceUrl =
    forceAuthManager.getInstanceUrl() ?:
    AppSettings.DEFAULT_FORCE_CONNECTED_APP.instanceUrl
val loyaltyAPIManager = LoyaltyAPIManager(
    auth = forceAuthManager,
    instanceUrl = mInstanceUrl,
    loyaltyClient = LoyaltyClient(forceAuthManager, mInstanceUrl)
)
```

3. To enroll an individual member, use this signature.

```
let membershipNumber = "1234567890"
let firstName = "John"
let lastName = "Doe"
let email = "john.doe@example.com"
let phone = "4157891234"
let emailNotification = true

loyaltyAPIManager.postEnrollment(
    firstName = firstName,
    lastName = lastName,
    email = email,
    additionalContactAttributes = mapof("phone" to phone),
    emailNotification = emailNotification,
```

```

        memberStatus = MemberStatus.ACTIVE,
        createTransactionJournals = true,
        transactionJournalStatementFrequency =
TransactionalJournalStatementFrequency.MONTHLY,
        transcationJournalStatementMethod =
TransactionalJournalStatementMethod.EMAIL,
        enrollmentChannel = EnrollmentChannel.EMAIL,
        canReceivePromotions = true,
        canReceivePartnerPromotions = true
    )

```

4. To retrieve the member benefits, use this signature.

```
loyaltyAPIManager.getMemberBenefits(memberId = null, membershipKey = "1234567890")
```

5. To retrieve the member profile, use this signature.

```
loyaltyAPIManager.getMemberProfile(memberId = null, membershipKey = "1234567890",
programCurrencyName = null)
```

6. To opt a member into a promotion, use this signature.

```
loyaltyAPIManager.enrollInPromotions(membershipNumber = "1234567890", promotionName =
"PromotionName")
```

7. To opt out a member from a promotion using promotion ID or promotion name, use this signature.

```
loyaltyAPIManager.unEnrollPromotion(membershipNumber = "1234567890", promotionName =
"PromotionName")
```

8. To retrieve loyalty member transactions, use this signature.

```
loyaltyAPIManager.getTransactions(membershipNumber = "1234567890", pageNumber = null,
journalTypeName = null, journalSubTypeName = null, periodStartDate = null,
periodEndDate = null)
```

9. To retrieve promotions for a loyalty member, use this signature.

```
loyaltyAPIManager.getEligiblePromotions(membershipKey = "1234567890", memberId = null)
```

10. To retrieve vouchers for a loyalty member, use this signature.

```
loyaltyAPIManager.getVouchers (
        membershipKey = "1234567890", vouchreStatus = null, pageNumber = 1,
productId = null,
        productCategoryId = null, productName = null, productCategoryName = null
    )

```

## LoyaltyAPIInterface

The `LoyaltyApiInterface` class defines the interface for creating, formatting, and sending REST requests to the Salesforce service. This class stores REST API call definitions for all Loyalty API.



Use this signature to define the `LoyaltyAPIInterface` class.

```
public interface LoyaltyApiInterface
```

## Method Summary

Modifier and Type	Method	Method Detail	Description
abstract Result<EnrollmentResponse>	<code>postEnrollment(@Url() String url, @Body() EnrollmentRequest json)</code>	<code>postEnrollment</code>  @POST () as a Result<EnrollmentResponse> of type @Url() String url, @Body() EnrollmentRequest json)	Creates a POST request for an individual enrollment for the loyalty program.
abstract Result<MemberProfileResponse>	<code>getMemberProfile(@Url() String url, @Query (value = "memberId") String memberId, @Query (value = "relationshipName") String relationshipName, @Query (value = "pageNo") String pageNo)</code>	<code>getMemberProfile</code>  @GET () as a Result<MemberProfileResponse> of type @Url() String url, @Query (value = "memberId") String memberId, @Query (value = "relationshipName") String relationshipName, @Query (value = "pageNo") String pageNo)	Retrieves profile details related to the member ID.
abstract Result<MemberBenefitsResponse>	<code>getMemberBenefits(@Url() String url, @Query (value = "relationshipName") String relationshipName)</code>	<code>getMemberBenefits</code>  @GET () as a Result<MemberBenefitsResponse> of type @Url() String url, @Query (value = "relationshipName") String relationshipName)	Retrieves benefits related to the member ID.
abstract Result<TransactionsResponse>	<code>getTransactions(@Url() String url, @Query (value = "pageNumber") Integer pageNumber, @Query (value = "journalType") String journalType, @Query (value = "journalStore") String journalStore, @Query (value = "journalCity") String journalCity, @Query (value = "periodStartDate") String periodStartDate, @Query (value = "periodEndDate") String periodEndDate)</code>	<code>getTransactions</code>  @GET () as a Result<TransactionsResponse> of type @Url() String url, @Query (value = "pageNumber") Integer pageNumber, @Query (value = "journalType") String journalType, @Query (value = "journalStore") String journalStore, @Query (value = "journalCity") String journalCity, @Query (value = "periodStartDate") String periodStartDate, @Query (value = "periodEndDate") String periodEndDate)	Retrieves transactions related to the member.
abstract Result<PromotionsResponse>	<code>getEligiblePromotions(@Url() String url, @Body() PromotionsRequest requestBody)</code>	<code>getEligiblePromotions</code>  @POST () as a Result<PromotionsResponse> of type @Url() String url, @Body() PromotionsRequest requestBody)	Retrieves promotions that a promotion is eligible to enroll for.

Modifier and Type	Method	Method Detail	Description
		<pre>@Body() PromotionsRequest requestBody)</pre>	
<code>abstract Result&lt;EnrollPromotionsResponse&gt;</code>	<code>enrollInPromotion(@Url() String url, @Body() PromotionsRequest requestBody)</code>	<pre>enrollInPromotion @POST() abstract Result&lt;EnrollPromotionsResponse&gt; enrollInPromotion(@Url() String url, @Body() PromotionsRequest requestBody)</pre>	Enrolls member in a promotion.
<code>abstract Result&lt;UnenrollPromotionsResponse&gt;</code>	<code>unenrollPromotion(@Url() String url, @Body() PromotionsRequest requestBody)</code>	<pre>unenrollPromotion @POST() abstract Result&lt;UnenrollPromotionsResponse&gt; unenrollPromotion(@Url() String url, @Body() PromotionsRequest requestBody)</pre>	Opts out a member from an available promotion.
<code>abstract Result&lt;VoucherResult&gt;</code>	<code>getVouchers(@Url() String url, @Query(value = "voucherStatus") String voucherStatus, @Query(value = "pageNumber") Integer pageNumber, @Query(value = "productId") String productId, @Query(value = "productCategoryId") String productCategoryId, @Query(value = "productName") String productName, @Query(value = "productCategoryName") String productCategoryName)</code>	<pre>getVouchers @GET() abstract Result&lt;VoucherResult&gt; getVouchers(@Url() String url, @Query(value = "voucherStatus") String voucherStatus, @Query(value = "pageNumber") Integer pageNumber, @Query(value = "productId") String productId, @Query(value = "productCategoryId") String productCategoryId, @Query(value = "productName") String productName, @Query(value = "productCategoryName") String productCategoryName)</pre>	Retrieves vouchers that are available to the program member.

## Loyalty Management Mobile SDK API References for Android

The Loyalty Management Mobile SDK for Android interacts with Salesforce Loyalty Management APIs and retrieves member benefits, transactions, profiles, promotions, vouchers, enrolls an individual member, opts a member out of a promotion.

Interact with the Loyalty Management APIs, including:

Resource	Description
<a href="#">/loyalty-programs/{LoyaltyProgramName}/individual-member-enrollments</a> on page 305	Individual Member Enrollments
<a href="#">/connect/loyalty/programs/{ProgramName}/program-process/{id}/Enroll</a> on page 335	Enroll for Promotions
<a href="#">services/apexrest/expense-analysis</a> on page 549	Expense Analysis
<a href="#">/connect/loyalty/programs/{ProgramName}/program-process/GetMemberPromotions</a>	Get Member Promotions
<a href="#">/connect/loyalty/member/{memberId}/memberbenefits</a>	Member Benefits
<a href="#">/loyalty-programs/{LoyaltyProgramName}/members</a>	Member Profile
<a href="#">/loyalty-programs/{ProgramName}/members/{relationshipName}/vouchers</a>	Member Vouchers
<a href="#">services/apexrest/receipt-status-updation</a> on page 567	Receipt Status Updation
<a href="#">services/apexrest/upload-receipt</a> on page 569	Receipt Upload
<a href="#">/connect/loyalty/programs/{ProgramName}/transaction-history</a>	Transaction History
<a href="#">/connect/loyalty/programs/{ProgramName}/program-process/{id}/OptOut</a>	Opt Out from a Promotion

### EDITIONS

Available in: all editions that have Loyalty Management enabled

## Loyalty Management Sample App for Mobile SDK

Loyalty Management Sample App for Mobile SDK is a configurable, native iOS and Android mobile app that delivers a personalized loyalty experience with every iOS and Android mobile device. Build the sample app for your loyalty program members to experience the loyalty and reward capabilities. The app provides a quick turnaround as opposed to a loyalty app that is built from scratch.

Loyalty Management Sample App for Mobile SDK offers a range of customization options. Customize the look and feel of your loyalty program, set up reward structures, and define the accrual and redemption rules.

### EDITIONS

Available in: all editions that have Loyalty Management enabled

### [Use the Loyalty Management Mobile SDK Sample App](#)

The MyNTORewards sample apps are a reference implementation of Loyalty Management Mobile SDK capabilities, and include the SDK's code, structure, and implementation. You can either implement the source code provided with the sample app in your existing code as-is or extend the capabilities of the code by modifying the code.

### [Walkthrough of the Sample App for Loyalty Management Mobile SDK](#)

Adam can enroll as a loyalty program member and can experience Loyalty Management on the mobile device. Let's look at how Adam uses the sample app for Loyalty Management Mobile SDK.

### [Customize the Android SDK](#)

After you build the Loyalty Management Mobile SDK for Android, you can customize the colors, fonts, and strings.

### [Customize the iOS SDK](#)

After you build the Loyalty Management Mobile SDK for iOS, you can customize the colors and fonts to suit your requirements. You can also replace the existing images and logos included in the SDK for the sample app.

## Use the Loyalty Management Mobile SDK Sample App

The MyNTORewards sample apps are a reference implementation of Loyalty Management Mobile SDK capabilities, and include the SDK's code, structure, and implementation. You can either implement the source code provided with the sample app in your existing code as-is or extend the capabilities of the code by modifying the code.

After ensuring you have the necessary development tools, dependencies, and the SDK installed, integrate the Loyalty Management Mobile SDK sample app code in your iOS or Android project:

1. Go to the location from where you downloaded the iOS or Android SDK, and select **SampleApps > MyNTORewards**.
2. From the Sample Apps project, copy the files that you want to reuse in your project.
3. Open an existing or new project in Xcode or Android Studio, and create a class or file.
4. Paste the files that you copied from the MyNTORewards sample apps.
5. Import any necessary dependencies that are used within.
6. Modify the code as per your requirements.
7. Test your app after you add the sample app code in iOS or Android.



**Example:** For example, consider the source code for the class, `CommunityMemberModel`:

```
package com.salesforce.loyalty.mobile.myntorewards.utilities

data class CommunityMemberModel (
    val firstName: String?,
    val lastName: String?,
    val email: String?,
    val loyaltyProgramMemberId: String?,
    val loyaltyProgramName: String?,
    val membershipNumber: String?
)
```

You can use the `CommunityMemberModel` class to represent the community member information as per your app. For example, you can create instances of this class to store member data:

```
val member = CommunityMemberModel (
    firstName = "Adam",
    lastName = "Smith",
    email = "adamsmith@example.com",
    loyaltyProgramMemberId = "123456",
    loyaltyProgramName = "NTO Insider Loyalty Program",
    membershipNumber = "78190"
)
```

You can modify the code provided with the sample app in the SDK to extract and store member data:

```
val firstName = member.firstName
val lastName = member.lastName
```

```
val email = member.email
val loyaltyProgramName = member.loyaltyProgramName
```

## Walkthrough of the Sample App for Loyalty Management Mobile SDK

Adam can enroll as a loyalty program member and can experience Loyalty Management on the mobile device. Let's look at how Adam uses the sample app for Loyalty Management Mobile SDK.

Capability	Adam's Journey
Join as loyalty program	Opens the sample app on a mobile device and signs up as a member of the NTO Rewards loyalty program.  Adam uses their Experience Cloud site email address to enroll in the NTO Insider program.
Review the membership details on the home page	Checks their current points balance, available promotions, vouchers, and badges.
Opt in for promotions	Reviews the list of promotions that are available to join and joins the Gold Tier Rejuvenation promotion.  As a promotion enrollment bonus, 500 reward points are credited to Adam's points balance.
Add items to a cart and place an order You can extend the checkout workflow based on your mobile app's requirements.	Adds a red jacket to the cart, proceeds to the checkout page, enters the payment details, and places the order.
Track order rewards	Views the transactions list on the home page to verify the points that have been credited for the placed order.
View vouchers received for orders	Views the list of vouchers to verify that the Holiday Season voucher has been issued as a reward for the placed order.
Scan receipts for third-party store purchases	Uses the QR code scanner that's available on the My Profile tab to scan the receipt of a purchase made at a third-party store.
View and play assigned games	Goes to the Game Zone tab, opens an assigned game, and plays the game.  After Adam plays the game, a game reward of 100 reward points is provided to Adam.

## Customize the Android SDK

After you build the Loyalty Management Mobile SDK for Android, you can customize the colors, fonts, and strings.

1. Customize the colors.
  - a. With your Sample Apps project open in Android Studio, open the **MyNTORewards > src > main > java > com > salesforce > loyalty > mobile > myntorewards > ui > theme > Color.kt** file.

- b. To add a color to the Android UI theme or as another app color, edit the `Color.kt` file.  
For example, to add blue as one of the app colors, open the `Color.kt` file, and under `//App Colors`, add the `val Blue = Color(0x0000FF33)` code.
    - c. Save the `Color.kt` file.
    - d. To add the new color in any of the component definitions, add an entry or replace an entry with the new color.  
For example, to replace the vibrant purple 40 color with blue, in any of the files, replace `color = VibrantPurple40` with `color = Blue`.
    - e. Clean and rebuild your project in Android Studio.
2. Add a font to the Android UI theme.
  - a. With your Sample Apps project open in Android Studio, go to the **MyNTORewards > src > main > res > font** folder.
  - b. In the font folder, paste a .ttf file with the font script that you want to add.
  - c. Open the **MyNTORewards > src > main > java > com > salesforce > loyalty > mobile > myntorewards > ui > theme > Type.kt** file.
  - d. Edit the `Type.kt` file, and add the file that you added in the font folder.  
For example, to add Arial as one of the fonts, edit the `Type.kt` file to add the sample code:
 

```
val font_arial = FontFamily(
    Font(R.font.arial)
)
```
  - e. Save the `Type.kt` file.
  - f. Add this font in any of the component definitions.  
For example, to replace the `sf_pro` font on the My Transactions page with Arial, replace `fontFamily = font_sf_pro` with `fontFamily = font_arial`.
  - g. Clean and rebuild your project in Android Studio.
3. Customize the UI labels or strings used in your Sample App.
  - a. Open the **MyNTORewards > src > main > res > values > strings.xml** file.
  - b. Edit any text included in the `<string>` tag.  
For example, to change the text Membership Number to Membership ID, replace all the strings that contain Membership Number with Membership ID.
  - c. Save the `strings.xml` file.
  - d. Clean and rebuild your project in Android Studio.
4. To replace an image, go to **MyNTORewards > src > main > res > drawable**, and replace the image.

## Customize the iOS SDK

After you build the Loyalty Management Mobile SDK for iOS, you can customize the colors and fonts to suit your requirements. You can also replace the existing images and logos included in the SDK for the sample app.

1. Customize the colors.

- a. With your Sample Apps project open in Xcode, select **MyNTORewards > MyNTORewards > Assets > Colors**.
- b. To modify an existing colorset, click **Any Appearance**, and under Color, edit the Hex value.
- c. To add the dark color, click **Dark**, and from Content, select a color.
- d. To add a color theme, open the **MyNTORewards > MyNTORewards > Misc > Color** file, and add a color theme to the sample code.

For example, to add blue color to the theme, use this code:

```
struct ColorTheme {let blue = Color("BlueColor")}
```

- e. To use a color in the app, use these sample codes as view or within view modifier:

```
ZStack {
    Color.theme.[color]
}
```

Or `background(Color.theme.[color])`

For example, to add Blue color to the theme, use this code as view or within view modifier:

```
ZStack {
    Color.theme.blue
}
```

Or `background(Color.theme.blue)`

2. Add your font to the iOS app.
  - a. [Add your font to the Xcode project](#).
  - b. Add your font to the **MyNTORewards > MyNTORewards > Info.plist** file.
3. Use the font that you added in the app.
  - a. With your Sample Apps project open in Xcode, select **MyNTORewards > MyNTORewards > UIComponents > FontStyles.swift** file.
  - b. Edit the file and add the location where you want to use the font. For example, to customize a footer text, use this sample code:
 

```
static var footerText: Font {
    return Font.custom("SFPro-Regular", size: 16)
}
```
  - c. In the Text view, call the font style as a new modifier with the sample code:
 

```
Text("Footer")
    .font(.footerText)
```
  - d. Save your changes.
4. To replace an image, go to **MyNTORewards > MyNTORewards > Assets > Images**. For example, to replace a logo, go to the **MyNTORewards > MyNTORewards > Assets > Icons** folder, and modify the logo file.

## Receipt Scanning with Loyalty Management Mobile SDK

---

Members of your loyalty program can scan and upload the purchase receipts of third-party store or site transactions on the Loyalty Management iOS or Android mobile app. After members upload their receipts, members are rewarded for eligible purchases, they can also view the accrued points on the Loyalty Management mobile app.

Watch this video to understand how you can enable members to earn rewards by purchasing your company's products from multibrand and third-party stores.

 [Watch a video](#)

If you aren't able to watch the video in full screen, open the video on a new tab: [🎥 Reward Members for Shopping at Multibrand \(Third-Party\) Stores.](#)

### [How Receipt Scanning Works](#)

Integrate Loyalty Management with receipt scanning to enhance your brand's member experience and streamline how members can earn loyalty points for their store purchases. The receipt scanning workflow is designed to automatically extract and organize essential information from scanned receipts and reward members for their store purchases.

### [Loyalty Management Receipt Scanner Kit Unlocked Package](#)

The Loyalty Management Receipt Scanner Kit unlocked package allows members to scan and submit receipts to receive loyalty points that they're purchases are eligible for. After the receipt is submitted, the package automatically creates transaction journals for the eligible receipt line items. The unlocked package contains a set of components, and developers can select the modular components that they want to use in their implementation. Unlocked packages are customizable and help you easily manage and reuse components across different projects.

### [Set Up Amazon Web Services](#)

To integrate the Loyalty Management Receipt Scanning Kit unlocked package with Amazon Web Services (AWS), create an S3 bucket that stores the receipt images, and create roles, policies, and users. Then, get the access key and secret ID to authenticate the access to AWS resources.

### [Set Up Salesforce Org With Loyalty Management Receipt Scanner Kit Unlocked Package](#)

To ensure that members can submit their receipts by using their Loyalty Management mobile app and get rewarded for eligible receipt line items, install and set up the Loyalty Management Receipt Scanner Kit Unlocked Package in your Salesforce org. Before you install the unlocked package, ensure that you have a loyalty program already set up in your org.

### [Receipt Scan and Upload on iOS and Android](#)

After you set up the Loyalty Management Receipt Scanner Kit unlocked package and integrate the Loyalty Management org with an Optical Character Recognition (OCR) software, members can upload and scan receipts by using the loyalty program's mobile app. Members are rewarded for with loyalty points for eligible receipt line items.

## How Receipt Scanning Works

Integrate Loyalty Management with receipt scanning to enhance your brand's member experience and streamline how members can earn loyalty points for their store purchases. The receipt scanning workflow is designed to automatically extract and organize essential information from scanned receipts and reward members for their store purchases.

### [Why Integrate Loyalty Management With a Receipt Scanner?](#)

Brands that don't own their sales channels can't accurately track customer purchases made in on third-party stores or sites. In such scenarios, the brand's loyalty program needs a receipt-scanning solution that can scan the items from a receipt, identify eligible items, and reward the points to the members.



### Receipt Scanning Workflow

The receipt scanning workflow describes how a program member uploads and scans a receipt, verifies the scanned result, and earns loyalty points.

### Receipt Evaluation

Members use their mobile phones to upload their purchase receipts. After a member uploads a receipt, they can review the receipt's scanned line items, and submit the scanned receipt. After a receipt is submitted, the eligible line items are processed as transaction journals and members are rewarded. You can define how a receipt is evaluated when you enable Receipt Scanning.

## Why Integrate Loyalty Management With a Receipt Scanner?

Brands that don't own their sales channels can't accurately track customer purchases made in on third-party stores or sites. In such scenarios, the brand's loyalty program needs a receipt-scanning solution that can scan the items from a receipt, identify eligible items, and reward the points to the members.

For Loyalty Management's receipt scanning integration, install the Loyalty Management Receipt Scanner Kit package and then integrate the package with Amazon Web Services (AWS) Textract Analyze Expense Optical Character Recognition (OCR). The integration helps convert text from receipts into a valid format. We can also integrate Loyalty Management with any other OCR software.

### EDITIONS

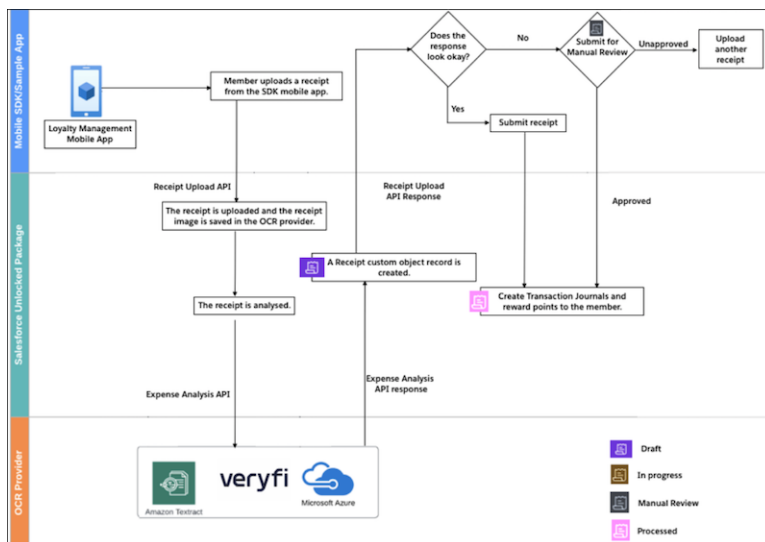
Available in: all editions that have Loyalty Management enabled



**Example:** Member shops at a shopping mall for their loyalty products. Then, the Member logs in to the mobile app and submits the receipt to earn points. The receipts are verified and the member is rewarded for the purchase of the eligible items.

## Receipt Scanning Workflow

The receipt scanning workflow describes how a program member uploads and scans a receipt, verifies the scanned result, and earns loyalty points.



- After a member uploads a receipt by using the Loyalty Management iOS or Android mobile app, the Receipt Upload API is automatically run to encode the receipt.
- The AnalyzeExpense API is then automatically executed for the encoded receipt, and the response creates a record to store the receipt data in an Amazon S3 bucket. The API scans the uploaded receipt and provides the details of the receipt as a response. The

response is saved in the custom object, Receipts, in the Loyalty Management app. The member then previews the scanned results on the mobile app, and submits the receipt. A transaction journal is created for each line item in the receipt.

- If the member submits the receipt for a manual review, the loyalty program manager or admin reviews, and approves or rejects the receipt.

If the program manager approves the receipt, the Receipt Status Updation API updates the status of the receipt to Processed, updates the reward points, and creates a transaction journal for each line item in the receipt. The RulesEngineResponseWrapper Apex class shows the response returned from Transaction Journal Execution API that includes the transaction journal details.

## Receipt Evaluation

Members use their mobile phones to upload their purchase receipts. After a member uploads a receipt, they can review the receipt's scanned line items, and submit the scanned receipt. After a receipt is submitted, the eligible line items are processed as transaction journals and members are rewarded. You can define how a receipt is evaluated when you enable Receipt Scanning.

Here are the factors for evaluating.

### EDITIONS

Available in: all editions that have Loyalty Management enabled

## Confidence Score

Define a confidence score to assess if the scanned receipt meets the criteria to extract the receipt information. A confidence value higher than the defined threshold value typically implies greater certainty in the accuracy of the receipt scanning result, while a lower value suggests more uncertainty in reading the receipt.

To understand how a receipt is evaluated based on the confidence score, consider these scenarios:

Business Scenario	Result	What Happens in this Scenario?
A member uploads a clean receipt and the receipt content is readable.	The confidence value for the receipt is higher than or equal to the threshold value.	Submit the receipt and get rewards for eligible items.
A member uploads a crumpled receipt or a receipt with poor image quality, and the receipt lines are partially readable.	The confidence value for the crumpled receipt is lower than the threshold value.	Submit the receipt for a manual review or capture the image again to ensure that it's readable.
A member uploads a crumpled or a torn receipt, and the mandatory fields on the receipt such as receipt number, date, and store name aren't readable.	The confidence value for the crumpled receipt is lower than the threshold value.	The receipt is rejected because the information required to process the receipt wasn't extracted.

## Ineligible Receipt Items

Configure the receipt scanning to show members the items from their receipt that aren't eligible for loyalty points. Typically, items that don't meet the criteria for earning loyalty points include certain excluded categories, products, or expired promotions.

## Required Receipt Fields

If any mandatory information is missing in a receipt uploaded by a member, the receipt is rejected. When you enable receipt scanning, Receipt Date is a mandatory field to be included in every receipt. You can select the other mandatory fields from these options:

- Receipt Number

- Store Name
- Postal Code

You can also configure additional mandatory fields on a receipt.

## Eligible Item Identifier Fields

When you enable receipt scanning, you can select fields that uniquely determine eligible items from a receipt. For example, if you select product name and product code as the eligible item identifier fields, the receipt is evaluated to determine the eligible items, and the results show the products that match the filter criteria with product name and product code.

Consider this example to understand how an eligible item is determined. Adam Smith submits a receipt by using the Loyalty Management mobile app.



The receipt is uploaded using the Receipt Upload API, and the Expense Analysis API provides the scanned results by using this JSON response:

```
{ "ExpenseDocuments": [
  { "SummaryFields": [
    { "ValueDetection": { "Text": "FROSTFLAKES SNOWFLAKES", "Type": { "Text": "NAME" } },
    { "ValueDetection": { "Text": "260.00", "Type": { "Text": "AMOUNT_PAID" } },
    { "ValueDetection": { "Text": "19-04-2018", "Type": { "Text": "INVOICE_RECEIPT_DATE" } },
    { "ValueDetection": { "Text": "38", "Type": { "Text": "INVOICE_RECEIPT_ID" } },
    { "ValueDetection": { "Text": "23BBNCA2901C1Z6", "Type": { "Text": "TAX_PAYER_ID" } },
    { "ValueDetection": { "Text": "23BBNCA2901C1Z6", "Type": { "Text": "VENDOR_GST_NUMBER" } },
    { "ValueDetection": { "Text": "247.62", "Type": { "Text": "SUBTOTAL" } },
    { "ValueDetection": { "Text": "6.19", "Type": { "Text": "TAX" } },
    { "ValueDetection": { "Text": "6.19", "Type": { "Text": "TAX" } },
    { "ValueDetection": { "Text": "260.00", "Type": { "Text": "TOTAL" } },
    { "ValueDetection": { "Text": "FROSTFLAKES SNOWFLAKES", "Type": { "Text": "VENDOR_NAME" } },
    { "ValueDetection": { "Text": "2001", "Type": { "Text": "OTHER" } },
    { "ValueDetection": { "Text": "01", "Type": { "Text": "OTHER" } },
    { "ValueDetection": { "Text": "06:17:34 PM", "Type": { "Text": "OTHER" } },
    { "ValueDetection": { "Text": "Q38", "Type": { "Text": "OTHER" } },
    { "ValueDetection": { "Text": "Cashier 1", "Type": { "Text": "OTHER" } },
    { "ValueDetection": { "Text": "QUICK SERVICE", "Type": { "Text": "OTHER" } },
    { "ValueDetection": { "Text": "Cash", "Type": { "Text": "OTHER" } }
  ]
}
```

```

"LineItemGroups": [
  "LineItems": [
    {"LineItemExpenseFields": [
      {"ValueDetection": {"Text": "THE PHAL"}, "Type": {"Text": "PRODUCT_NAME"}},
      {"ValueDetection": {"Text": "95.24"}, "Type": {"Text": "UNIT_PRICE"}},
      {"ValueDetection": {"Text": "1.00"}, "Type": {"Text": "QUANTITY"}},
      {"ValueDetection": {"Text": "95.24"}, "Type": {"Text": "PRICE"}},
      {"ValueDetection": {"Text": "THE PHAL 95.24 1.00 95.24"}, "Type": {"Text": "EXPENSE_ROW"}}
    ]},
    {"LineItemExpenseFields": [
      {"ValueDetection": {"Text": "NATURAL"}, "Type": {"Text": "PRODUCT_NAME"}},
      {"ValueDetection": {"Text": "76.19"}, "Type": {"Text": "UNIT_PRICE"}},
      {"ValueDetection": {"Text": "1.00"}, "Type": {"Text": "QUANTITY"}},
      {"ValueDetection": {"Text": "76.19"}, "Type": {"Text": "PRICE"}},
      {"ValueDetection": {"Text": "NATURAL 76.19 1.00 76.19"}, "Type": {"Text": "EXPENSE_ROW"}}
    ]},
    {"LineItemExpenseFields": [
      {"ValueDetection": {"Text": "FIGS"}, "Type": {"Text": "PRODUCT_NAME"}},
      {"ValueDetection": {"Text": "76.19"}, "Type": {"Text": "UNIT_PRICE"}},
      {"ValueDetection": {"Text": "1.00"}, "Type": {"Text": "QUANTITY"}},
      {"ValueDetection": {"Text": "76.19"}, "Type": {"Text": "PRICE"}},
      {"ValueDetection": {"Text": "FIGS 76.19 1.00 76.19"}, "Type": {"Text": "EXPENSE_ROW"}}
    ]}
  ]}

```

The loyalty program admin has selected Price as the eligible item identifier field, and so the receipt line item's response shows eligible items by comparing the line item's price with the product price. The eligible items in the receipt are: The Phal, Natural, and Figs.

## Loyalty Management Receipt Scanner Kit Unlocked Package

The Loyalty Management Receipt Scanner Kit unlocked package allows members to scan and submit receipts to receive loyalty points that they're purchases are eligible for. After the receipt is submitted, the package automatically creates transaction journals for the eligible receipt line items. The unlocked package contains a set of components, and developers can select the modular components that they want to use in their implementation. Unlocked packages are customizable and help you easily manage and reuse components across different projects.

### EDITIONS

Available in: all editions that have Loyalty Management enabled

### [Loyalty Management Receipt Scanner Kit Unlocked Package Components](#)

The Loyalty Management Receipt Scanner Kit unlocked package contains a set of components such as custom objects, fields, Apex classes, Lightning web components, and more to facilitate easier distribution of these bundled components.

## Loyalty Management Receipt Scanner Kit Unlocked Package Components

The Loyalty Management Receipt Scanner Kit unlocked package contains a set of components such as custom objects, fields, Apex classes, Lightning web components, and more to facilitate easier distribution of these bundled components.

### EDITIONS

Available in: all editions that have Loyalty Management enabled

### Custom Objects

The package consists of custom objects that store the details of the scanned receipts.

Object	What the Object Stores
Receipt	Saves the details of the scanned and submitted receipts
Receipt Scanning Error	Saves the details of the errors that occur while scanning receipts.

## Permission Set, List View, Page Layout, and Tab

The Receipt Scanning permission set enables loyalty program members to access the Receipt Scanning feature.

Loyalty program managers can review all submitted receipts on the Receipts tab on the Loyalty Management app. In Setup, the Receipt Scanning setting enables your loyalty program members to scan receipts on their Loyalty Management iOS and Android mobile apps, and to earn loyalty points for eligible items.

## Custom Settings

The package consists of a custom setting that stores the values that are required to connect Loyalty Management with an Optical Character Recognition (OCR) software.

Setting	What the Setting Stores
API Endpoint Details	<p>The API Gateway endpoint URL that is used to access and interact with</p> <ul style="list-style-type: none"> <li>• The APIs</li> <li>• The bucket details that store the objects used in the AWS endpoint</li> <li>• The API Named Credential that provides the API Gateway endpoint's URL</li> <li>• The authentication settings for Salesforce to make callouts to AWS</li> </ul>

## Custom Label

The package consists of a custom label that stores the values that are required to reward non-qualifying points.

Setting	What the Setting Stores
PrimaryNQPCurrencyName	This label stores the primary NQP currency name. To replace this currency name with your program's primary NQP currency, from the Setup page, search and find <b>Custom Label</b> , and then edit and replace the label.

## Apex Classes

The unlocked package contains Apex classes to run the Expense Analysis API and Receipt Status Update API for submitted receipts, and to show the scan results on the Loyalty Management mobile app.

Apex Class	What the Class Does
DMLManager	Manages Data Manipulation Language (DML) statements to insert, update, merge, delete, and restore data in Salesforce.
ExpenseAnalysis	Manages the request to process the uploaded receipt.
ProcessTransactionJournal	Creates transaction journals for each line item of a scanned receipt and calls the Rules Engine API to process the transaction journals.
ReceiptScanningConfigController	Manages the receipt scanning configuration.
ReceiptScanningControllerUtils	Manages the common functionality methods.
ReceiptScanningResponseWrapper	Analyzes the eligible and ineligible items in the receipt after a receipt is uploaded and returns a response of line items for the scanned receipt.
ReceiptStatusUpdate	Updates the status of receipts by sending REST requests to a specific location or endpoint.
ReceiptUpload	Manages the request to upload the receipt and creates a request to analyze the uploaded receipt.
ReceiptWrapper	Represents the data structure of an AWS response.
RulesEngineResponseWrapper	Analyzes the response returned from the Rules Engine API that includes transaction journal details.

## Custom Metadata Types

The unlocked package contains custom metadata types to create and store the objects and fields related to receipt scanning.

Custom Metadata Type	What the Custom Metadata Type Does	Examples
Accepted Date Format	Represents the configuration of the accepted date format and the store name fields related to the receipt.	26/02/2022
Required Receipt Fields	Represents the configuration of the mandatory fields to be captured from a scanned receipt.	Receipt Date
Receipt Scanning	Represents the configuration details related to the receipt scanning feature.	Confidence score, named credential, receipt scanning activity, and show ineligible receipt items

## Apex Trigger

The unlocked package contains apex triggers to update the status of a receipt and to automatically assign the Receipt Scanning permission set to Experience Cloud users.

Apex Trigger	What the Apex Trigger Does
ReceiptUpdateTrigger	Creates transaction journals, and processes the journals by using the Rule Engine API.
UserReceiptScanTrigger	<p>Automatically assigns the Receipt Scanning permission set to all the Experience Cloud profile users, after the admin installs the unlocked package.</p> <p>For existing loyalty program members, the admin has to manually assign the Receipt Scanning permission set to the members' Experience Cloud users.</p>

## Customizations

Admins can customize their receipt scanning implementation by extending the component provided by the Loyalty Management Receipt Scanner Kit unlocked package. The admin can add additional checks that decide which elements must be available in a receipt for it to be eligible to be rewarded.



**Example:** For example, if the loyalty program manager wants to add a filter logic that decides which products details must be available in a receipt, the admin can customize these package components:

- Apex classes
  - ReceiptScanningResponseWrapper
  - ReceiptScanningControllerUtils
  - receiptScanningDetail
  - Eligible Item Identifier Field custom metadata type

## Set Up Amazon Web Services

To integrate the Loyalty Management Receipt Scanning Kit unlocked package with Amazon Web Services (AWS), create an S3 bucket that stores the receipt images, and create roles, policies, and users. Then, get the access key and secret ID to authenticate the access to AWS resources.

### 1. [Create Your Amazon S3 Bucket](#)

Create a bucket to store your receipt images and any other data in AWS.

### 2. [Create an IAM Role](#)

Create your Identity and Access Management (IAM) role to control the access to your Amazon S3 bucket. The IAM role ensures the security and functioning of AWS resources.

### 3. [Attach a Policy to the IAM Role](#)

After you create an Identity and Access Management (IAM) role for your S3 bucket, create a policy and attach the policy to the IAM role. Policies define the actions for IAM roles and users.

### 4. [Get Your AWS ARN](#)

After the Identity and Access Management (IAM) role is created, and a policy is attached to the role, you can copy the unique identifiers of AWS resources, Amazon Resource Names (ARN). Use the ARN when you create the API gateway.

5. [Create an IAM User](#)

An Identity and Access Management (IAM) user manages the access and permissions for the users and applications within your AWS account. Create an IAM user and assign permissions to the IAM user to independently maintain security and access control in your AWS Management Console.

6. [Create Access Key and Secret](#)

After you create an IAM user, create an access key for the user and download the access key and secret. Use the access keys and secrets to authenticate and authorize programmatic access to AWS resources.

7. [Create an API Gateway](#)

Create an API Gateway that serves as an entry point for all incoming API requests.

8. [Get the Invoke URL](#)

After you create the API and select a deployment stage, deploy the API so that members can upload their receipts and get loyalty points. This invoke URL is used to create the API Endpoint Details custom setting and represents the Remote Site URL for your AWS API Gateway endpoint.

## Create Your Amazon S3 Bucket

Create a bucket to store your receipt images and any other data in AWS.

1. Log in to your AWS Management Console.
2. From the Services menu, find and select **S3**.
3. Click **Create bucket**.
4. In AWS Region, select the region where you want the bucket to reside.
5. Enter a name for your bucket.
6. Click **Create bucket**.

### EDITIONS

Available in: all editions that have Loyalty Management enabled

## Create an IAM Role

Create your Identity and Access Management (IAM) role to control the access to your Amazon S3 bucket. The IAM role ensures the security and functioning of AWS resources.

1. Log in to your AWS Management Console.
2. From the Services menu, find and select **IAM**.
3. From the navigation menu, click **Roles**.
4. Click **Create role**.
5. From the Service or use case dropdown list, select **API Gateway**, and then click **Next**.
6. On the Add permissions page, click **Next**.
7. Enter the role name and description, and then click **Create role**.

### EDITIONS

Available in: all editions that have Loyalty Management enabled



## Attach a Policy to the IAM Role

After you create an Identity and Access Management (IAM) role for your S3 bucket, create a policy and attach the policy to the IAM role. Policies define the actions for IAM roles and users.


1. Log in to your AWS Management Console.
2. From the Services menu, find and select **IAM**.
3. From the navigation menu, click **Roles**.
4. Click the IAM role that you created.
5. Under Permissions, from Add permissions, select **Attach policies**.  
If you see the Create policy button instead of the Attach policies button, click Create policy and follow the subsequent steps.
6. From the navigation menu, under Access management, click **Policies**, and then click **Create policy**.
7. From Service, select **S3**.
8. In the Actions allowed list, find and select the PutObject write action.
9. Under Resources, click **Add ARNs**.
10. Specify the ARNs.
  - a. Enter your S3 bucket name as the resource bucket name.
  - b. In Resource object name, select **Any object name**.
  - c. Click **Add ARNs**.
11. Click **Next**.
12. On the Review and create page, enter your policy name and description, and then click **Create policy**.  
The policy is created, you can now attach this policy to the IAM role.
13. On the Policies page, find and select your policy.
14. Click **Entities Attached**.
15. Under Attached as a permissions policy, click **Attach**, and select the IAM role that you created.
16. Click **Attach policy**.
17. From the navigation menu, under Access management, click **Roles**, and then select the role that you created.
18. Under Permissions, from Add permissions, select **Attach Policies**.
19. Select the **AdministratorAccess** option, and then click **Add permissions**.

### EDITIONS

Available in: all editions that have Loyalty Management enabled

## Get Your AWS ARN

After the Identity and Access Management (IAM) role is created, and a policy is attached to the role, you can copy the unique identifiers of AWS resources, Amazon Resource Names (ARN). Use the ARN when you create the API gateway.

1. Log in to your AWS Management Console.
2. From the Services menu, find and select **IAM**.
3. From the navigation menu, click **Roles**.
4. Click the IAM role that you created.
5. Under Summary, click  to copy the ARN.

### EDITIONS

Available in: all editions that have Loyalty Management enabled

## Create an IAM User

An Identity and Access Management (IAM) user manages the access and permissions for the users and applications within your AWS account. Create an IAM user and assign permissions to the IAM user to independently maintain security and access control in your AWS Management Console.

1. Log in to your AWS Management Console.
2. From the Services menu, find and select **IAM**.
3. From the navigation menu, click **Users**.
4. Click **Create user**.
5. Enter a username, and then click **Next**.
6. Under Permissions options, click **Attach policies directly**.
7. Under Permissions policies, enter *extract*.
8. Select **AmazonTextractFullAccess**, and find and select **AmazonS3ReadOnlyAccess**.
9. Click **Next**.
10. Click **Create user**.

### EDITIONS

Available in: all editions that have Loyalty Management enabled

## Create Access Key and Secret

After you create an IAM user, create an access key for the user and download the access key and secret. Use the access keys and secrets to authenticate and authorize programmatic access to AWS resources.

1. Log in to your AWS Management Console.
2. From the Services menu, find and select **IAM**.
3. From the navigation menu, click **Users**.
4. Select the user that you created.
5. Under Summary, click **Create access key**.
6. Select **Application running outside AWS**, and then click **Next**.
7. Enter a description to define the purpose of the key, and then click **Create access key**.
8. To ensure that you don't lose the access credentials, download the CSV file that contains the access key and secret.

### EDITIONS

Available in: all editions that have Loyalty Management enabled

## Create an API Gateway

Create an API Gateway that serves as an entry point for all incoming API requests.

1. Log in to your AWS Management Console.
2. From the Services menu, find and select **API Gateway**.
3. Click **Create API**.
4. Under REST API, click **Build**.
5. Enter a name and description for the new API, and then click **Create API**.
6. Click **Create resource**.
7. Enter *{bucket}* as the resource name.

### EDITIONS

Available in: all editions that have Loyalty Management enabled

8. Select **CORS (Cross Origin Resource Sharing)**, and then click **Create resource**.
9. To create another resource for bucket, select **bucket**, and then click **Create resource**.
10. Enter `{filename}` as the resource name.
11. Select **CORS (Cross Origin Resource Sharing)**, and then click **Create resource**.
12. To create a method under filename, select **filename**, and then click **Create method**.
13. For the Method type, select **PUT**.
14. Select **AWS service** as the integration type.
15. From AWS Region, select the region of your S3 bucket.
16. From AWS service, select **Simple Storage Service (S3)**.
17. Select **PUT** as the HTTP method.
18. Select **Use path override** as the action type.
19. In Execution role, paste the ARN for the IAM role that you created earlier.
20. Click **Create method**.
21. On the PUT-Method Execution page, click **Integration request**.
22. Click **Edit**.
23. Expand URL path parameters, and click **Add path**, and enter these details:
  - Name: `bucket`
  - Mapped from: `method.request.path.bucket`
24. Save your changes.
25. Under Integration request, click **Edit**, and then expand URL path parameters.
26. Click **Add path** again, and enter these details:
  - Name: `filename`
  - Mapped from: `method.request.path.filename`
27. Save your changes.
28. From the navigation menu, click **API settings**.
29. To allow all the media types, under Binary media types, click **Manage media types**.
30. Click **Add binary media type**, and enter `*/*`.
31. Save your changes.
32. Similarly, create the Get and Delete methods under `/{bucket}/{filename}`.


## Get the Invoke URL

After you create the API and select a deployment stage, deploy the API so that members can upload their receipts and get loyalty points. This invoke URL is used to create the API Endpoint Details custom setting and represents the Remote Site URL for your AWS API Gateway endpoint.

1. Log in to your AWS Management Console.
2. From the Services menu, find and select **API Gateway**.
3. Click the name of the API that you created.

### EDITIONS

Available in: all editions that have Loyalty Management enabled

4. Click **Deploy API**.
5. Select the deployment stage.  
If you don't have a stage, select **[New Stage]** and enter the stage name and the deployment description.
6. Click **Deploy**.  
The deployment takes a couple of minutes to be completed.
7. Under Stage details, click , and copy the Invoke URL.

## Set Up Salesforce Org With Loyalty Management Receipt Scanner Kit Unlocked Package

To ensure that members can submit their receipts by using their Loyalty Management mobile app and get rewarded for eligible receipt line items, install and set up the Loyalty Management Receipt Scanner Kit Unlocked Package in your Salesforce org. Before you install the unlocked package, ensure that you have a loyalty program already set up in your org.

1. [Install the Loyalty Management Receipt Scanner Kit Unlocked Package](#)  
Install the Loyalty Management Receipt Scanner Kit unlocked package in the org that contains the loyalty program for members to upload their purchase receipts by using the Loyalty Management mobile app.
2. [Assign Permission Set to Existing Users](#)  
After admins install the Loyalty Management Receipt Scanner Kit unlocked package in their org, new users are automatically assigned the Receipt Scanning permission set. The Receipt Scanning permission set provides users access to manage the capabilities of the Loyalty Management Receipt Scanner Kit unlocked package. Assign existing users who need the permissions to manage and work with the capabilities of the package the Receipt Scanning permission set.
3. [Assign Permission Set for Members to Access Receipt Scanner](#)  
For your members to access the Receipt Scanner feature on their mobile devices, assign the Receipt Scanning permission set to the Experience Cloud users. Assign permission set to the members to ensure that Apex classes are automatically run when members upload a receipt. After you assign the permission set, modify the Receipts object to provide read and edit permissions.
4. [Create a Named Credential](#)  
Connect Amazon Web Services (AWS) with the Loyalty Management Receipt Scanner Kit unlocked package. Make callout requests to AWS from Salesforce after a receipt is uploaded. The AnalyzeExpense API then scans the receipts to analyze and interpret the information from the receipt.
5. [Create a Connected App](#)  
Create a connected app for Amazon Web Services (AWS) to connect with the Loyalty Management Receipt Scanner Kit unlocked package. This connected app ensures that the Expense Analysis API runs after the receipts are uploaded and creates transaction journals for the submitted receipts.
6. [Add a Loyalty Program Name](#)  
The Loyalty Management Receipt Scanner Kit unlocked package has a custom label to store the name of your loyalty program.
7. [Add API Endpoint Details in Custom Setting](#)  
The Loyalty Management Receipt Scanner Kit unlocked package has a custom setting to store the API Gateway endpoint URL and bucket details from your Amazon Web Services (AWS) account.
8. [Create a Remote Site Setting](#)  
Before any Apex callout can call the API Gateway endpoint, register the API Gateway endpoint on the Remote Site Settings page. Remote site settings authorize your org to access external websites.

## 9. [Enable Receipt Scanning](#)

Enable receipt scanning in the org where the Loyalty Management Receipt Scanner Kit unlocked package is installed. Members can then use their mobile device to upload receipts and get loyalty points.

## Install the Loyalty Management Receipt Scanner Kit Unlocked Package

Install the Loyalty Management Receipt Scanner Kit unlocked package in the org that contains the loyalty program for members to upload their purchase receipts by using the Loyalty Management mobile app.

1. From your loyalty org, click [Loyalty Management Receipt Scanner Kit Unlocked Package](#).
2. Enter your Salesforce credentials.
3. Select the audience for your package.
4. Click **Install**.

After the unlocked package is installed, the Receipt Scanning permission set is automatically assigned to the Experience Cloud users' of new loyalty program members and Apex classes are automatically run when members upload a receipt. Members can now use the Loyalty Management app on their mobile devices to submit receipts and earn loyalty points.

For existing loyalty program members, you've to manually assign the Receipt Scanning permission set to the members' Experience Cloud users.

### EDITIONS

Available in: all editions that have Loyalty Management and Receipt Scanning enabled

## Assign Permission Set to Existing Users

After admins install the Loyalty Management Receipt Scanner Kit unlocked package in their org, new users are automatically assigned the Receipt Scanning permission set. The Receipt Scanning permission set provides users access to manage the capabilities of the Loyalty Management Receipt Scanner Kit unlocked package. Assign existing users who need the permissions to manage and work with the capabilities of the package the Receipt Scanning permission set.

1. From Setup, in the Quick Find box, enter *Permission Sets*, and then select **Permission Sets**.
2. Click **Receipt Scanning > Manage Assignments > Add Assignment**.
3. Select the existing users must manage and work with the receipt scanning capabilities.
4. Click **Next > Assign**.

### EDITIONS

Available in: all editions that have Loyalty Management and Receipt Scanning enabled

### USER PERMISSIONS

To assign a permission set:

- Assign Permission Sets

To assign a permission set license:

- Manage Users

## Assign Permission Set for Members to Access Receipt Scanner

For your members to access the Receipt Scanner feature on their mobile devices, assign the Receipt Scanning permission set to the Experience Cloud users. Assign permission set to the members to ensure that Apex classes are automatically run when members upload a receipt. After you assign the permission set, modify the Receipts object to provide read and edit permissions.

1. From Setup, in the Quick Find box, enter *User*, and then select **Users**.
2. Select the user record of the Loyalty Customer Community User.
3. Click **Permission Set Assignments > Edit Assignments**.
4. Move the **Receipt Scanning** permission set to the Enabled Permission Sets list.
5. Save your changes.
6. From Setup, in the Quick Find box, enter *Permission Sets*, and then select **Permission Sets**.
7. Click **Receipt Scanning**.
8. Click **Object Settings**.
9. From the object settings, click **Receipts**, and then click **Edit**.
10. For the ReceiptScanner\_\_ReceiptID\_\_c field, select the Read and Edit permissions.
11. Save your changes.

## Create a Named Credential

Connect Amazon Web Services (AWS) with the Loyalty Management Receipt Scanner Kit unlocked package. Make callout requests to AWS from Salesforce after a receipt is uploaded. The AnalyzeExpense API then scans the receipts to analyze and interpret the information from the receipt.

1. From Setup, in the Quick Find box, enter *Named*, and then select **Named Credentials**.
2. From the dropdown, select **New Legacy**.
3. Enter these details:
  - Label: Enter a name for the legacy named credential that's shown in your org.
  - Name: Enter a unique identifier that's used to refer to this legacy-named credential from callout definitions and through the API.
  - URL: Enter `https://textract.[REGION].amazonaws.com`. For example, `https://textract.ap-south-1.amazonaws.com/`.
4. Select **Named Principal** as the identity type.
5. Select **AWS Signature Version 4** as the authentication protocol, and enter these details from the CSV file that's downloaded from AWS:
  - AWS Access Key ID: Enter the access key ID. For example, `AKIAQO45CVNMD4NEU3VB`.
  - Access Secret: Enter the secret access key. For example, `51rC7gCfPF+RfF8GWxZnjBw++ACT3ESm5BYwH+jS`
  - AWS Region: Enter the region that you selected in AWS. For example, `ap-south-1`.
  - AWS Service: Enter `textract` as the AWS service.
6. Save your changes.

### EDITIONS

Available in: all editions that have Loyalty Management and Receipt Scanning enabled

### USER PERMISSIONS

To assign a permission set:

- Assign Permission Sets

To assign a permission set license:

- Manage Users

### EDITIONS

Available in: all editions that have Loyalty Management and Receipt Scanning enabled

### USER PERMISSIONS

To view named credentials:

- View Setup and Configuration

To create, edit, or delete named credentials:

- Manage Named Credentials OR Customize Applications

## Create a Connected App

Create a connected app for Amazon Web Services (AWS) to connect with the Loyalty Management Receipt Scanner Kit unlocked package. This connected app ensures that the Expense Analysis API runs after the receipts are uploaded and creates transaction journals for the submitted receipts.

1. From Setup, in the Quick Find box, enter *certificate*, and then select **Certificate and Key Management**.
2. Under Certificates, click **Create Self-Signed Certificate**.
3. Enter a label and unique name for the certificate.
4. Save the new certificate.
5. Click **Download Certificate**.
6. From Setup, in the Quick Find box, enter *Apps*, and then select **App Manager**.
7. Click **New Connected App**, and fill these fields.
  - Connected App Name: Enter *Manage Authentication*
  - API Name: Keep the default API name that's automatically filled based on the connected app name
  - Contact Email: Enter your email address
  - Description: Enter *Manage how external users are authenticated when they connect with Salesforce applications*
8. Under API (Enable OAuth Settings), select **Enable OAuth Settings**.
9. In the Callback URL, enter *https://login.salesforce.com/services/oauth2/token/callback*.
10. Select **Use digital signatures**.
11. Click **Choose file**, and upload the certificate that you downloaded.
12. In the Selected OAuth Scopes section, move the **Manage user data via APIs (api)** and **Perform requests at any time (refresh\_token, offline\_access)** values to Selected OAuth Scopes.
13. Save the connected app.
14. From Setup, in the Quick Find box, enter *Connected Apps*, and then select **Manage Connected Apps**.
15. Next to Manage Authentication, click **Edit**.
16. Select **Admin approved users are Pre-authorized** as the permitted users.
17. From Timeout Value, select **24 hours**.
18. Save your changes.
19. Click **Manage Authentication**.
20. Under Profiles, click **Manage Profiles**.
21. Select **System Administrator**, and save your changes.
22. In the Quick Find box, enter *custom*, and then select **Custom Settings**.
23. For the Manage Authenticate Setting whose namespace prefix is ReceiptScanner, select **Manage**, and then select **New**.
24. Under Manage Authenticate Setting Information, enter these values.
  - Name: Enter *Manage Authentication*

### EDITIONS

Available in: all editions that have Loyalty Management enabled

### USER PERMISSIONS

To create connected apps:


- Customize Application
- AND
- Modify All Data OR  
Manage Connected Apps

- Audience: Based on your environment, enter one of the values: `https://test.salesforce.com/` or `https://login.salesforce.com/`
- Certificate: Enter the certificate name, without the extension
- Consumer Key: Enter the consumer key. To get the consumer key, from the App Manager setup, for the Manage Authentication app, select **View**. Under API (Enable OAuth Settings), click **Manage Consumer Details**, enter your credentials, and then copy the consumer key.
- Subscriber: Enter the username of the Salesforce admin.
- Token Endpoint: Based on your environment, enter one of the values: `https://test.salesforce.com/services/oauth2/token` or `https://login.salesforce.com/services/oauth2/token`

25. Save your changes.

## Add a Loyalty Program Name

The Loyalty Management Receipt Scanner Kit unlocked package has a custom label to store the name of your loyalty program.

 **Note:** If you installed the Loyalty Management Receipt Scanner Kit unlocked package in a B2C Retail Industry Template org, then the loyalty program name is already set up.

1. In the Quick Find box, enter `custom label`, and then select **Custom Label**.
2. Next to Loyalty Program Name, click **Edit**.
3. Enter your loyalty program name.
4. Save your changes.

## Add API Endpoint Details in Custom Setting

The Loyalty Management Receipt Scanner Kit unlocked package has a custom setting to store the API Gateway endpoint URL and bucket details from your Amazon Web Services (AWS) account.

1. In the Quick Find box, enter `custom settings`, and then select **Custom Settings**.
2. Next to API Endpoint Details, click **Manage**.
3. Click **New**.
4. Enter these values.
  - API Gateway Endpoint: From your AWS Account, enter the Invoke URL until the stage name. For example, if the Invoke URL is

`https://baznrtuyv3.execute-api.us-east-2.amazonaws.com/abc`, where `abc` is the stage name, enter the endpoint URL as `https://baznrtuyv3.execute-api.us-east-2.amazonaws.com`.

- Bucket Name: From your AWS account, enter the S3 bucket name.
- Select Disable Duplicate Check if you want members to submit the same receipt multiple times.

### EDITIONS

Available in: all editions that have Loyalty Management enabled

### USER PERMISSIONS

Create, edit, or delete custom labels:

- Customize Application

### EDITIONS

Available in: all editions that have Loyalty Management and Receipt Scanning enabled

### USER PERMISSIONS

To manage, create, edit, and delete custom settings:

- Customize Application



5. Save your changes.  
The location of the endpoint is automatically selected.

## Create a Remote Site Setting

Before any Apex callout can call the API Gateway endpoint, register the API Gateway endpoint on the Remote Site Settings page. Remote site settings authorize your org to access external websites.

1. In the Quick Find box, enter *Remote*, and then select **Remote Site Settings**.
2. Click **New Remote Site**.
3. Enter *AWS Gateway Remote Site* as the remote site name.
4. Enter the Invoke URL of your Amazon Web Services account.  
For example, a sample Invoke URL looks like this:  
`https://*****.execute-api.us-east-2.amazonaws.com`
5. Save your changes.

## Enable Receipt Scanning

Enable receipt scanning in the org where the Loyalty Management Receipt Scanner Kit unlocked package is installed. Members can then use their mobile device to upload receipts and get loyalty points.

1. From Setup, in the Quick Find box, enter *Lightning*, and then select **Lightning App Builder**.
2. Next to the Receipt Scanning app, click **Edit**.
3. From the Setup menu, click **Edit Page**.
4. Click **Activation**.
5. To change the icon, click **Change**, and then assign another logo for Receipt Scanning.
6. Click the Lightning Experience tab, and then click **Loyalty Management**.
7. In the Loyalty Management section, drag the Receipt Scanning app based on where you want the app to appear.
8. Save your changes.
9. From the App Launcher, enter *Receipt*, and then select **Receipt Scanning**.
10. Under Named Credential, click **Assign Named Credential**.
11. Select the named credential that you created for the Loyalty Management Receipt Scanning Kit unlocked package to connect with AWS.
12. Read and agree to the terms and conditions, and then save your changes.
13. Turn on Receipt Scanning.
14. Enter the confidence value at which the receipt is eligible for submission.
15. To show ineligible items for a member after a receipt is scanned and analyzed, turn on Ineligible Receipt Items.
16. Under Required Receipt Fields, select the fields that you want to include in a scanned receipt.

### EDITIONS

Available in: all editions that have Loyalty Management and Receipt Scanning enabled

### USER PERMISSIONS

To configure remote settings:

- Customize Application or Modify All Data

### EDITIONS

Available in: all editions that have Loyalty Management and Receipt Scanning enabled

### USER PERMISSIONS

To create and save Lightning Pages in the Lightning App Builder:

- Customize Application


To view Lightning Pages in the Lightning App Builder:

- View Setup and Configuration

Select the Receipt Number field if you want to uniquely identify each receipt that's uploaded.

**17.** Under Eligible Item Identifier Fields, select the fields that identify the eligible items from a receipt.

**18.** Save your changes.

 **Example:** If you select Receipt Number as a required receipt field, the members can upload only the receipts that contain a receipt number. If you select Product Name as an eligible item identifier field, you earn loyalty points only for the items with matching product names in the Products object.

## Receipt Scan and Upload on iOS and Android

After you set up the Loyalty Management Receipt Scanner Kit unlocked package and integrate the Loyalty Management org with an Optical Character Recognition (OCR) software, members can upload and scan receipts by using the loyalty program's mobile app. Members are rewarded for with loyalty points for eligible receipt line items.

Members can use the loyalty program's mobile app to upload purchase receipts from third-party partner stores and sites. To get the latest mobile app SDK, developers can clone and import the latest Loyalty Management Mobile SDKs for iOS and Android.

### [Considerations Before You Upload a Receipt](#)

Members can upload receipts of products purchased at third-party stores, and get rewards for eligible purchases.

### [Upload and Submit a Receipt](#)

Members can upload receipts from their third-party store purchases that are less than 5 MB, and get loyalty points for eligible receipt line items. After the receipt is scanned, members can review the line items detected by the Expense Analysis API and then submit the receipt.

### [View Submitted Receipts](#)

Loyalty program members can view the submitted receipts and line items from their loyalty management mobile app after uploading the receipts, and view the loyalty points credit for a processed receipt.

## Considerations Before You Upload a Receipt

Members can upload receipts of products purchased at third-party stores, and get rewards for eligible purchases.

To ensure the receipt upload and scan process is seamless, check these conditions before you upload a scanned receipt.

- The receipt is legible, and not torn or crumpled.
- The size of the receipt file is less than 5 MB.
- The receipt is valid and accurate, and contains this data:
  - purchase date
  - receipt number
  - store name and address,
  - product information
  - quantity purchased
  - price of a unit
  - total price
  - any other information that's mandatory to process the receipt.

### EDITIONS

The Loyalty Management mobile app is available in: all editions that have Loyalty Management and Receipt Scanning enabled


- The receipt isn't duplicated.
- The receipt's date format matches the store's date format that's specified in the Accepted Date Format custom metadata type.

The custom metadata type supports these date formats:

- mm-dd-yyyy
- dd-mm-yyyy
- yyyy-mm-dd
- mm.dd.yyyy
- dd.mm.yyyy
- yyyy.mm.dd
- mm/dd/yyyy
- dd/mm/yyyy
- yyyy/mm/dd

## Upload and Submit a Receipt

Members can upload receipts from their third-party store purchases that are less than 5 MB, and get loyalty points for eligible receipt line items. After the receipt is scanned, members can review the line items detected by the Expense Analysis API and then submit the receipt.

1. On the home screen of the iOS or the Android Loyalty Management mobile app, tap .
2. Tap **New**, and provide access to your photo library.
3. Tap the camera, and click the photo of a receipt or tap the gallery icon and upload a receipt from your mobile device.
4. Tap **Upload**.  
The receipt is processed and the results show eligible and ineligible items from the receipt.
5. Select an option:
  - If the processed receipt results are accurate, tap **Submit**, and then tap **Done**.  
The receipt is submitted and the screen shows the number of loyalty points credited for the receipt.
  - To rescan or upload another receipt, tap **Try Again**.
  - If your receipt doesn't list all eligible items, tap **Request a Manual Review**.  
Enter a reason for the manual request, and tap **Submit**.

### EDITIONS

The Loyalty Management mobile app is available in: all editions that have Loyalty Management and Receipt Scanning enabled

### [What Happens When Members Upload Receipts](#)

Members can upload receipts of their third-party store purchases and after they upload the receipts, the Expense Analysis API scans the receipts and reviews the eligibility of the receipt.

### [What Happens When Members Submit Receipts](#)

After members submit receipts, they can view the submitted receipts and the line items for which they are rewarded from their loyalty management mobile app.

### [What Happens When Members Submit Receipts for Manual Review](#)

If the eligible items of a receipt don't show all items or if there's an incorrect evaluation of loyalty points for the receipt, members can submit the receipt for a manual review.

## What Happens When Members Upload Receipts

Members can upload receipts of their third-party store purchases and after they upload the receipts, the Expense Analysis API scans the receipts and reviews the eligibility of the receipt.

Here's what happens in the Loyalty Management Receipt Scanner Kit unlocked package org when members upload their receipts.

- The ReceiptUpload Apex class runs the Receipt Upload API to upload the receipt.
- Then, the ReceiptScanningControllerUtils Apex class runs the Expense Analysis API to scan receipts uploaded by members.
- The ExpenseAnalysis class runs the Expense Analysis API to analyze the receipt.
- The API scans the receipts and provides a response, which is stored in the ReceiptScanningResponseWrapper class. The response consists of the store details, receipt ID, number, and date, and the detected line item details. The response also shows a confidence value, if applicable.

### EDITIONS

Available in: all editions that have Loyalty Management and Receipt Scanning enabled

## What Happens When Members Submit Receipts

After members submit receipts, they can view the submitted receipts and the line items for which they are rewarded from their loyalty management mobile app.

Here's what happens in your Loyalty Management Receipt Scanner Kit unlocked package org after members submit their receipts.

- The ReceiptStatusUpdate Apex class runs the Receipt Status Updation API, to update the status of the receipt to Processed.
- The ProcessTransactionJournal Apex class runs the [Transaction Journal Execution Business API](#), which creates and processes transaction journals.
- The RulesEngineResponseWrapper class deserializes the response from the Rules Engine API that includes transaction journal details. The response provides loyalty points, which are shown in the mobile app.
- The Receipt Status Updation API creates and processes receipt records for each line item.

### EDITIONS

The Loyalty Management mobile app is available in: all editions that have Loyalty Management and Receipt Scanning enabled


## What Happens When Members Submit Receipts for Manual Review

If the eligible items of a receipt don't show all items or if there's an incorrect evaluation of loyalty points for the receipt, members can submit the receipt for a manual review.

Here's what happens when the receipt is submitted for manual review.

- The ReceiptStatusUpdate Apex class runs the Receipt Status Updation API, to update the status of the receipt to Manual Review.
- The ReceiptUpdateTrigger updates the receipt record.

To view the receipt in the Loyalty Management app, loyalty program managers go to the App Launcher, and find and select **Receipts**. After the admin or loyalty program manager manually completes the review, the Receipts screen on the mobile app shows the status of the receipt. If the receipt is manually approved, the member gets loyalty points for the eligible items.


 **Tip:** Admins can also add the Receipts related list on the Loyalty Program Member record. This allows program managers to review all the receipts that are pending review for a loyalty program member.

### EDITIONS

The Loyalty Management mobile app is available in: all editions that have Loyalty Management and Receipt Scanning enabled

## View Submitted Receipts

Loyalty program members can view the submitted receipts and line items from their loyalty management mobile app after uploading the receipts, and view the loyalty points credit for a processed receipt.

1. On the home screen of the iOS or the Android Loyalty Management mobile app, tap .
2. On the Receipts screen, view these details:
  - Receipt number
  - Receipt date
  - Receipt total
  - Loyalty points credited for a processed receipt
  - Receipt status for an unprocessed receipt
3. Tap the receipt and view the details of each eligible and ineligible detected line item including:
  - Item name
  - Quantity purchased
  - Unit price
  - Total price
  - Receipt image
4. To download the receipt image, tap **Receipt Image**, and then tap **Download Original Receipt**.

### EDITIONS

The Loyalty Management mobile app is available in: all editions that have Loyalty Management and Receipt Scanning enabled

## Loyalty Management Mobile SDK and Sample App Support

---

Loyalty Management Mobile SDK is an open-source product. If you have any questions or feedback on the mobile SDK, post your questions or feedback on the Salesforce Stack Exchange site and use the `loyaltymobile` hashtag. If you find a bug with either the SDK or the sample app, raise a ticket on GitHub.

## Upgrade Loyalty Management Mobile SDK for iOS

---

For a major upgrade, clone and import the latest version of the SDK. For minor or incremental patch updates, update the Loyalty Management Mobile SDK to the latest package version.

- To update the Loyalty Management Mobile SDK for iOS to the latest package version, in Xcode, click **File > Swift Packages > Update to Latest Package Versions**.
- For a major upgrade of Loyalty Management Mobile SDK for iOS, clone and import the latest version of the Loyalty Management Mobile SDK for iOS.

### EDITIONS

Available in: all editions that have Loyalty Management enabled

#### SEE ALSO:

[Clone and Import Loyalty Management Mobile SDK for iOS](#)

[Clone and Import Loyalty Management Mobile SDK for iOS](#)

## CHAPTER 12 Gamification Mobile SDK


### In this chapter ...

- [Supported Versions of Tools and Components](#)
- [Gamification Mobile SDK for iOS](#)
- [Gamification Mobile SDK for Android](#)

Enhance brand engagement by providing gamification features on Android and iOS mobile devices. Use the Gamification mobile Software Development Kit (SDK) for Android and iOS to build custom mobile applications with unique game reward experiences. Or, use the SDK as an external dependency in your existing Loyalty Management Mobile SDK, and extend the `MYNTORewards` sample app to include games that a loyalty member can play to win rewards.

Watch this video to understand how to design and run games.

 [Watch a video](#)

If you aren't able to watch the video in full screen, open the video in a new tab:  [Gamify Member Engagement with Loyalty Management](#).

## Supported Versions of Tools and Components

The Gamification Mobile SDK works on iOS 15.0+ and Android 8.0+ versions. Install the tools and components required to set up the Gamification Mobile SDK.

### EDITIONS

Available in: all editions that have Loyalty Management enabled

### iOS

Tool or Component	Supported Version	Installation Details
Swift Version	5.7+	Installed by Xcode
Xcode	14.0+	Install from the App Store
iOS SDK	15.0+	Installed by Xcode
Swift Package Manager	5.7+	Included in Swift

### Android

Tool or Component	Supported Version	Installation Details
Android Studio Electric Eel	2022.1.1+	Install from the Android Developer site
Android SDK and APIs	8.0+ (API level 26+)	Install from the Android Developer site
Android SDK Tools	8.0+	Install from the Android Developer site
Android Virtual Device	8.0+	Install from Android Studio
Kotlin	1.8.0+	Installed by Android Studio

## Gamification Mobile SDK for iOS

Clone and import the Gamification Mobile iOS SDK to integrate the gamification feature in a mobile app.

### EDITIONS

Available in: all editions that have Loyalty Management enabled

#### [Clone and Import Gamification Mobile SDK for iOS](#)

Clone the Gamification Mobile iOS SDK GitHub repository to copy the code base and revision history of the repository to your local computer. After you clone the repository, import Gamification Mobile iOS SDK in an Xcode project.

#### [Integrate Gamification Mobile iOS SDK into Loyalty Management Mobile iOS SDK](#)

Import the Gamification Mobile iOS SDK into the existing Loyalty Management Mobile iOS SDK by adding it as an external dependency.

#### [Gamification Mobile SDK iOS Classes](#)

The iOS classes organize the Gamification Mobile SDK's functionalities and data structures. When developing mobile apps using the SDK, these classes provide components, services, or features with which you can interact.

### [Gamification Mobile SDK API Reference for iOS](#)


The Gamification Mobile iOS SDK interacts with the Salesforce Gamification APIs, and retrieves games assigned to a loyalty member, and the rewards won by the member after playing the game.

### [Game Zone UI Customization for iOS](#)

After you build the Gamification Mobile SDK for iOS, you can customize the game zone UI components.

## Clone and Import Gamification Mobile SDK for iOS

Clone the Gamification Mobile iOS SDK GitHub repository to copy the code base and revision history of the repository to your local computer. After you clone the repository, import Gamification Mobile iOS SDK in an Xcode project.

1. Go to the [Gamification Mobile SDK iOS repository](#).
2. To copy the repository URL to your clipboard, click **Code**, and then click .
3. Open Terminal, and go to the directory where you want to clone the repository.
4. Enter `git clone https://github.com/loyaltysampleapp/GamificationMobileSDK-iOS`.
5. Press Enter and wait for the repository to clone.
6. Open the project where you want to import `GamificationMobileSDK-iOS` in Xcode.
7. Go to **File > Swift Packages > Add Package Dependency**.
8. Click **Add Local**, and then select the local `GamificationMobileSDK-iOS` folder.
9. Select a version, and click **Add Package** and on the confirmation window, click **Add Package**.
10. To import the `GamificationMobileSDK-iOS` module into your Swift file, open the swift file where you want to use `GamificationMobileSDK-iOS`, and to the first line of code, add:

```
import GamificationMobileSDK-iOS
```

You can now use the classes, structures, and protocols in the `GamificationMobileSDK-iOS` module to manage and customize the gamification features in your mobile application.

### EDITIONS

Available in: all editions that have Loyalty Management enabled

## Integrate Gamification Mobile iOS SDK into Loyalty Management Mobile iOS SDK

Import the Gamification Mobile iOS SDK into the existing Loyalty Management Mobile iOS SDK by adding it as an external dependency.

1. Open the `LoyaltyMobileSDK-iOS` project in Xcode.
2. Go to **File > Swift Packages > Add Package Dependency**.
3. Enter the repository

URL: `https://github.com/loyaltysampleapp/GamificationMobileSDK-iOS`, or if you've cloned the repository, click **Add Local**, and then select the local `GamificationMobileSDK-iOS` folder.

4. Select a version, and click **Add Package** and on the confirmation window, click **Add Package**.
5. To import the `GamificationMobileSDK-iOS` module into your Swift file, open the swift file where you want to use `GamificationMobileSDK-iOS`, and to the first line of code, add:

```
import GamificationMobileSDK-iOS
```

### EDITIONS

Available in: all editions that have Loyalty Management enabled



You can now use the classes, structures, and protocols in the `GamificationMobileSDK-iOS` module to extend and manage the gamification features in the `MyNTORewards` sample app.

## Gamification Mobile SDK iOS Classes

The iOS classes organize the Gamification Mobile SDK's functionalities and data structures. When developing mobile apps using the SDK, these classes provide components, services, or features with which you can interact.

### [APIManager](#)

The `APIManager` class manages requests related to gamification, and invokes the gamification REST APIs. Use this class to interact with the Salesforce gamification APIs and retrieve member games, and game rewards.

### [GamificationForceClient](#)

The `GamificationForceClient` class handles network requests using the `GamificationForceAuthenticator` protocol.

### [GamificationForceAuthenticator](#)

The `GamificationForceAuthenticator` protocol defines the necessary methods for handling access tokens in the Salesforce API.

### EDITIONS

Available in: all editions that have Loyalty Management enabled

## APIManager

The `APIManager` class manages requests related to gamification, and invokes the gamification REST APIs. Use this class to interact with the Salesforce gamification APIs and retrieve member games, and game rewards.

The Gamification Mobile iOS SDK requires a valid access token to interact with the Salesforce gamification APIs. To provide a valid access token, pass an instance of `GamificationForceAuthenticator` protocol in your `APIManager` class at the time of initialization. The `APIManager` class manages authentication using the provided instance of `GamificationForceAuthenticator`.

Use this initializer to initialize the `GamificationRemoteRepository` class.

```
public init(auth: GamificationForceAuthenticator, instanceURL: String, forceClient:
GamificationForceClient) {
    self.auth = auth
    self.instanceURL = instanceURL
    self.forceClient = forceClient
}
```

Parameter	Type	Description
<code>auth</code>	<code>GamificationForceAuthenticator</code>	An instance of the <code>GamificationForceAuthenticator</code> protocol.
<code>instanceUrl</code>	<code>String</code>	The URL of your Salesforce org.
<code>forceClient</code>	<code>GamificationForceClient</code>	An instance of the <code>GamificationForceClient</code> class.

## Methods

### getGames

Retrieve games for the specified participant ID. This method returns an instance of the `GameModel` object.

```
public func getGames(
    participantId: String,
    version: String = APIVersion.defaultVersion,
    devMode: Bool = false,
    mockFileName: String = "GetGames_Success"
) async throws -> GameModel
```

Parameter	Type	Description
<code>participantId</code>	String	The ID of a loyalty program member, an account, or a contact whose eligible games are to be retrieved.
<code>version</code>	String	The API version.
<code>mockFileName</code>	String	The name of the file that has the sample response.
<code>devMode</code>	Boolean	Indicates whether the method is run in the developer mode ( <code>true</code> ) or not ( <code>false</code> ).

### getGameReward

Retrieve the details of the reward that's provided to a participant after the game play. This method returns an instance of the `PlayGameModel` object.

```
public func playGame(
    gameParticipantRewardId: String,
    version: String = APIVersion.defaultVersion,
    devMode: Bool = false,
    mockFileName: String = "PlayGame_Success"
) async throws -> PlayGameModel
```

Parameter	Type	Description
<code>gameParticipantRewardId</code>	String	The ID of the participant's game reward won by the participant whose details are to be fetched.
<code>version</code>	String	The API version.
<code>mockFileName</code>	String	The name of the file that has the sample response.
<code>devMode</code>	Boolean	Indicates whether the method is run in the developer mode ( <code>true</code> ) or not ( <code>false</code> ).

## GamificationForceClient

The `GamificationForceClient` class handles network requests using the `GamificationForceAuthenticator` protocol.

This class uses a `NetworkManager` instance to handle network requests and data processing.

Use this initializer to initialize the `GamificationForceClient` class.

```
public init(auth: GamificationForceAuthenticator, forceNetworkManager: NetworkManagerProtocol
           = NetworkManager.shared)
```

Parameter	Type	Description
<code>auth</code>	<code>GamificationForceAuthenticator</code>	An instance of the <code>GamificationForceAuthenticator</code> protocol.
<code>forceNetworkManager</code>	<code>NetworkManagerProtocol</code>	A <code>NetworkManager</code> instance.

## Methods

### fetchLocalJson

Returns a decoded JSON response result from the given local JSON file.

```
public func fetchLocalJson<T: Decodable>(type: T.Type, file: String, bundle: Bundle =
Bundle.publicModule)
```

Parameter	Type	Description
<code>type</code>	<code>T.Type</code>	The model to be used by the JSON decoder.
<code>file</code>	<code>String</code>	The file name of a local JSON file.
<code>bundle</code>	<code>Bundle</code>	The app bundle name.

### fetch

Returns a decoded JSON response result.

```
public func fetch<T: Decodable>(type: T.Type, with request: URLRequest, urlSession:
URLSession = .shared)
```

Parameter	Type	Description
<code>type</code>	<code>T.Type</code>	The model to be used by the JSON decoder.
<code>request</code>	<code>URLRequest</code>	The URL request to be executed by the URL session.
<code>urlSession</code>	<code>URLSession</code>	An instance of <code>URLSession</code> to execute the request.

## GamificationForceAuthenticator

The `GamificationForceAuthenticator` protocol defines the necessary methods for handling access tokens in the Salesforce API.

### Methods

#### `getAccessToken()`

Returns a valid access token as a string if available, otherwise null.

```
func getAccessToken()
```

#### `grantAccessToken`

Returns a refreshed or newly granted access token as a string.

```
func grantAccessToken()
```

## Gamification Mobile SDK API Reference for iOS

The Gamification Mobile iOS SDK interacts with the Salesforce Gamification APIs, and retrieves games assigned to a loyalty member, and the rewards won by the member after playing the game.

### [Assigned Games](#)

Get a list of eligible, expired, and played games, such as Spin the Wheel, or Scratch card, for a loyalty program member, an account, or a contact.

### [Game reward](#)

Get the details of the reward that's provided to a participant after the participant plays a Spin the Wheel, or Scratch Card game.

### EDITIONS

Available in: all editions that have Loyalty Management enabled.

## Assigned Games

Get a list of eligible, expired, and played games, such as Spin the Wheel, or Scratch card, for a loyalty program member, an account, or a contact.

### Resource

```
getGames(participantId:version:devMode:mockFileName:)
```

### Signature

```
public func getGames(
    participantId: String,
    version: String = APIVersion.defaultVersion,
    devMode: Bool = false,
    mockFileName: String = "GetGames_Success"
) async throws -> GameModel
```

## Input Parameters

Parameter	Type	Description	Required or Optional
devMode	Boolean	Indicates whether the method is run in the developer mode ( <code>true</code> ) or not ( <code>false</code> ).	Optional
mockFileName	String	The name of the file that has the sample response.	Optional
participantId	String	The ID of a loyalty program member, an account, or a contact whose eligible games are to be retrieved.	Required
version	String	Version of the API. The request body is available in version 60.0 and later.	Optional

## Return Value

A `GameModel` instance.

```
public struct GameModel : Codable
```

## Sample Output

```
{
  "message":null,
  "gameDefinitions":[
    {
      "description":"Play Spin The Wheel to get amazing rewards",
      "endDate":"2024-07-31T19:00:00.000Z",
      "gameRewards":[
        {
          "color":"FF5733",
          "description":"Get 10% Off Voucher",
          "imageUrl":null,
          "name":"10 % Off",
          "rewardType":"Voucher",
          "rewardValue":null
        },
        {
          "color":"000000",
          "description":"Loyalty Points",
          "imageUrl":null,
          "name":"1000 Reward Points",
          "rewardType":"LoyaltyPoints",
          "rewardValue":"100.0"
        }
      ]
    }
  ]
}
```

```

        "color":"762E2E",
        "description":"Better Luck Next Time",
        "imageUrl":null,
        "name":"No Reward",
        "rewardType":"NoReward",
        "rewardValue":null
    }
],
"participantGameRewards":[
    {
        "expirationDate":"2024-01-04T20:00:00.000Z",
        "gameParticipantRewardId":"3PRSB0000000hjJ4AQ",
        "gameRewardId":null,
        "issuedRewardReference":null,
        "sourceActivity":null,
        "status":"YetToReward"
    }
],
"name":"Spin The Wheel",
"startDate":"2023-10-01T19:00:00.000Z",
"timeoutDuration":10,
"type":"SpintheWheel"
},
{
    "description":"Play Scratch card to get amazing rewards",
    "endDate":"2024-07-31T19:00:00.000Z",
    "gameRewards":[
        {
            "color":"",
            "description":"Get 10% Off Voucher",
            "imageUrl":null,
            "name":"10 % Off",
            "rewardType":"Voucher",
            "rewardValue":null
        },
        {
            "color":"",
            "description":"Better Luck Next Time",
            "imageUrl":null,
            "name":"No Reward",
            "rewardType":"NoReward",
            "rewardValue":null
        }
    ],
    "participantGameRewards":[
        {
            "expirationDate":"2024-01-04T20:00:00.000Z",
            "gameParticipantRewardId":"2ABSB0000000hjJ3AM",
            "gameRewardId":"2GRSB0000000Caf4AB",
            "issuedRewardReference":"01VSB000000032j2AA",
            "sourceActivity":"0c85i0000009c6D",
            "status":"Rewarded"
        }
    ],

```

```

        "name": "Scratch Card",
        "startDate": "2023-10-01T19:00:00.000Z",
        "timeoutDuration": 20,
        "type": "Scratchcard"
    }
],
"status": true
}

```

## Output Parameters

Property Name	Type	Description	Filter Group and Version	Available Version
message	String	Message that provides the reason why a list of games weren't found for the participant.	Small, 60.0	60.0
gameDefinitions	GameDefinition[]	List of game definitions associated with the games assigned to the member. A game definition has the details of a game, including: <ul style="list-style-type: none"> <li>The list of possible rewards for the game.</li> <li>The participant game reward records that contain details of the game reward the participant has won for each game play, or can win after playing the game.</li> </ul>	Small, 60.0	60.0
status	Boolean	Indicates whether the request succeeded ( <code>true</code> ) or failed ( <code>false</code> ).	Small, 60.0	60.0

**Table 11: GameDefinition**

Property Name	Type	Description	Filter Group and Version	Available Version
description	String	Description of the game.	Small, 60.0	60.0
endDate	String	Date until which participants are eligible to play the game.	Small, 60.0	60.0
gameDefinitionId	String	ID of the game definition associated with the game.	Small, 60.0	60.0
gameRewards	GameReward []	List of possible rewards for a game.	Small, 60.0	60.0
name	String	Name of the game definition.	Small, 60.0	60.0
participantGameRewards	ParticipantGameReward[]	List of participant game reward records that contain the details of the game reward the	Small, 60.0	60.0

Property Name	Type	Description	Filter Group and Version	Available Version
		participant has won for each game play, or can win after playing the game.		
startDate	String	Date from which participants are eligible to play the game.	Small, 60.0	60.0
timeoutDuration	Integer	Duration of the game in seconds before the game session times out.	Small, 60.0	60.0
type	String	Type of the game. Possible values are: <ul style="list-style-type: none"> <li>Scratchcard</li> <li>SpintheWheel</li> </ul>	Small, 60.0	60.0

Table 12: GameReward

Property Name	Type	Description	Filter Group and Version	Available Version
color	String	Color of the segment in the Spin the Wheel game that represents the game reward.	Small, 60.0	60.0
description	String	Description of the game reward.	Small, 60.0	60.0
imageUrl	String	Image URL of the segment in the Spin the Wheel game that represents the game reward.	Small, 60.0	60.0
name	String	Name of the game reward.	Small, 60.0	60.0
gameRewardId	String	ID of the game reward.	Small, 60.0	60.0
rewardType	String	Type of the game reward. Possible values are: <ul style="list-style-type: none"> <li>CustomReward</li> <li>LoyaltyPoints</li> <li>NoReward</li> <li>Raffle</li> <li>Voucher</li> </ul>	Small, 60.0	60.0
rewardValue	String	Value of the voucher, or the number of points to credit as a game reward to participants.	Small, 60.0	60.0



**Table 13: ParticipantGameReward**

Property Name	Type	Description	Filter Group and Version	Available Version
expirationDate	String	Date until which the game reward is valid.	Small, 60.0	60.0
gameParticipantRewardId	String	ID of the participant game reward object.	Small, 60.0	60.0
gameRewardId	String	ID of the game reward won. This field is null for a game yet to be played.	Small, 60.0	60.0
issuedRewardReference	String	Voucher ID, or the transaction journal ID associated with the voucher, or the loyalty points rewarded to the participant. This field is null for a game yet to be played.	Small, 60.0	60.0
sourceActivity	String	ID of the promotion, or the transaction journal that is associated with the activity because of which the participant was eligible to play the game. Examples of source activities include purchasing a product, and reviewing on social media.  This field is null for a game yet to be played.	Small, 60.0	60.0
status	String	Status of the reward. Possible values are: <ul style="list-style-type: none"> <li>Expired</li> <li>NoReward</li> <li>Rewarded</li> <li>YetToReward</li> </ul>	Small, 60.0	60.0

## Game reward

Get the details of the reward that's provided to a participant after the participant plays a Spin the Wheel, or Scratch Card game.

## Resource

```
playGame(gameParticipantRewardId:version:devMode:mockFileName:)
```

## Signature

```
public func playGame(
    gameParticipantRewardId: String,
    version: String = APIVersion.defaultVersion,
    devMode: Bool = false,
    mockFileName: String = "PlayGame_Success"
) async throws -> PlayGameModel
```

## Input Parameters

Parameter	Type	Description	Required or Optional
devMode	Boolean	Indicates whether the method is run in the developer mode ( <code>true</code> ) or not ( <code>false</code> ).	Optional
mockFileName	String	The name of the file that has the sample response.	Optional
gameParticipantRewardId	String	The ID of the participant's game reward won by the participant whose details are to be fetched.	Required
version	String	Version of the API. The request body is available in version 60.0 and later.	Optional

## Return Value

A `PlayGameModel` instance.

```
public struct PlayGameModel: Codable
```

## Sample Output

```
{
  "message":null,
  "gameReward":[
    {
      "color":"01CD6C",
      "description":"Loyalty Points",
      "gameRewardId":"3GRSB0000000Caf4AE",
      "imageUrl":null,
      "issuedRewardReference":"01VSB000000032j2AA",
      "name":"1000 Reward Points",
      "rewardType":"LoyaltyPoints",
      "rewardValue":"100.0"
    }
  ],
  "status":true
}
```

## Output Parameters

Property Name	Type	Description	Filter Group and Version	Available Version
message	String	Error message that provides the reason why the details of the participant reward weren't found.	Small, 60.0	60.0
gameReward	PlayGameReward []	Details of the game reward won by the participant.	Small, 60.0	60.0
status	Boolean	Indicates whether the request succeeded ( <code>true</code> ) or failed ( <code>false</code> ).	Small, 60.0	60.0

**Table 14: PlayGameReward**

Property Name	Type	Description	Filter Group and Version	Available Version
color	String	Color of the segment in the Spin the Wheel game that represents the game reward.	Small, 60.0	60.0
description	String	Description of the game reward.	Small, 60.0	60.0
gameRewardId	String	ID of the game reward won by the participant.	Small, 60.0	60.0
imageUrl	String	Image URL of the segment in the Spin the Wheel game that represents the game reward.	Small, 60.0	60.0
issuedRewardReference	String	Voucher ID, or the transaction journal ID associated with the voucher, or the loyalty points rewarded to the participant.	Small, 60.0	60.0
name	String	Name of the game reward.	Small, 60.0	60.0
rewardType	String	Type of the game reward. Possible values are: <ul style="list-style-type: none"> <li>• CustomReward</li> <li>• LoyaltyPoints</li> <li>• NoReward</li> <li>• Raffle</li> <li>• Voucher</li> </ul>	Small, 60.0	60.0
rewardValue	String	Value of the voucher, or the number of points to credit as a game reward to participants.	Small, 60.0	60.0

## Game Zone UI Customization for iOS

After you build the Gamification Mobile SDK for iOS, you can customize the game zone UI components.

The game zone UI components are available in the

`LoyaltyMobileSDK-iOS/SampleApps/MyNTORewards/MyNTORewards/GameZone` package.

### Spinner Customization

Customize the spinner components, such as the wheel size, frame width and color, wheel segments, spin button, and more in the `FortuneWheelView.swift` file. Here are all the parameters you can modify.

```
VStack {
    HStack {
        Button {
            // Invalidating the timer to avoid unintended navigation because
of the timer

            timer?.invalidate()
            isSpinning ? nil : dismiss()
        } label: {
            Image("ic-backarrow")
        }
        .padding(.leading, 20)
        .padding(.bottom, 10)
        .frame(width: 30, height: 30)

        Spacer()
    }
    ZStack {
        Color.theme.accent
        VStack {
            VStack(spacing: 10) {
                Text(gameDefinitionModel?.name ??
StringConstants.Gamification.spinaWheelHeaderLabel)
                    .font(.gameHeaderTitle)

                Text(gameDefinitionModel?.description ??
StringConstants.Gamification.spinaWheelSubHeaderLabel)
                    .font(.gameHeaderSubTitle)
            }
            .padding(30)

            Spacer()
            ZStack {
                // Fortune Wheel Segments
                ZStack {
                    if let colors: [Color] = viewModel.getWheelColors(gameModel:
gameDefinitionModel),
                        let labels: [LocalizedStringKey] =
gameDefinitionModel?.gameRewards.map({LocalizedStringKey($0.name)}) {
                        ForEach(0..

```

```

Double(index + 1)) - 90.0
                WheelSegment(startAngle: startAngle,
                              endAngle: endAngle,
                              color: colors[index],
                              label: labels[index])
            }
        }
    }
    .overlay {
        ZStack {
            Circle()
                .strokeBorder(Color.theme.fortuneWheelStrokeColor,
lineWidth: 5)
                .frame(width: 316, height: 316)
            Circle()
                .strokeBorder(Color.theme.fortuneWheelSecondaryStrokeColor, lineWidth: 1)
                .frame(width: 307, height: 307)
        }
        .rotationEffect(Angle(degrees: rotationAngle))
        .animation(isSpinning ? Animation.easeOut(duration:
gameDefinitionModel?.timeoutDuration ?? 15)
            .delay(0)
            .repeatForever(autoreverses: false) : .default, value:
isSpinning)

        // Triangle Arrow Indicator
        ZStack {
            Triangle()
                .fill(Color.white)
                .frame(width: 70, height: 70) // Increase these
dimensions by the stroke width
                .offset(y: -40)

            Triangle()
                .fill(Color.theme.wheelIndicatorBackground)
                .frame(width: 60, height: 60)
                .offset(y: -35)
        }

        // Spin Button
        Button {
            playGame()
        } label: {
            userStartedSpinning ?

Text(StringConstants.Gamification.tapSpinButtonLabel).foregroundColor(Color.theme.wheelIndicatorBackground)
:

Text(StringConstants.Gamification.tapSpinButtonLabel).foregroundColor(.white)

        }
        .frame(width: 70, height: 70)

```

```

        .background(Color.theme.wheelIndicatorBackground)
        .clipShape(Circle())
        .overlay(Circle().stroke(Color.white, lineWidth: 5))
        .disabled(userStartedSpinning) // Disable the button when
spinning

    }
    .frame(width: 300, height: 300)

    Spacer()

    VStack(spacing: 20) {
        Text((StringConstants.Gamification.tapSpinaWheeltoPlayLabel))

        .font(.gameDescTitle)
        Text(StringConstants.Gamification.spinaWheelBodyLabel)
        .font(.gameDescText)
        .multilineTextAlignment(.center)
        .frame(width: 258)
    }
    Spacer()
}
.foregroundColor(.white)
}
.cornerRadius(15, corners: [.topLeft, .topRight])
.edgesIgnoringSafeArea(.bottom)
.fullScreenCover(isPresented: $showAlertForError) {
    Spacer()
    ProcessingErrorView(message: "Oops! Something went wrong while processing
the request. Try again.")
    Spacer()
    Button {
        timer?.invalidate()
        isSpinning ? nil : dismiss()
    } label: {
        Text(StringConstants.Receipts.tryAgainButton)
        .frame(maxWidth: .infinity)
    }
    .buttonStyle(.borderedProminent)
    .longFlexibleButtonStyle()
}
}
}
}

```

For example, to change the spin wheel outer frame width and height, open the `LoyaltyMobileSDK-iOS/SampleApps/MyNIORewards/MyNIORewards/GameZone/Views/FortuneWheel/FortuneWheelView.swift` file in Xcode, and replace `Circle() frame(width: 316, height: 316)` with `Circle() frame(width: 300, height: 300)`.

```

overlay {
    ZStack {
        Circle()
        .strokeBorder(Color.theme.fortuneWheelStrokeColor,
lineWidth: 5)

```

```

                .frame(width: 316, height: 316)
                Circle()

.strokeBorder(Color.theme.fortuneWheelSecondaryStrokeColor, lineWidth: 1)
                .frame(width: 307, height: 307)
            }
        }
    }

```

To change the spin button background color to blue, open the `LoyaltyMobileSDK-iOS/SampleApps/MyNTORewards/MyNTORewards/Misc/Color.swift` file, and inside the `ColorTheme` structure, replace `let wheelIndicatorBackground = Color("wheelIndicatorBackgroundColor") // #23475F` with `let wheelIndicatorBackground = Color.blue`.

## Scratch Card Customization

You can customize the scratch card components, such as the box, canvas, wrapper text, and more in the `ScratchCanvasView.kt` file. Here are all the properties you can customize:

```

private var backButton: some View {
    HStack {
        Button {
            // Invalidating the timer to avoid unintended navigation because of the
timer
            timer?.invalidate()
            dismiss()
        } label: {
            Image("ic-backarrow")
        }
        .padding(.leading, 20)
        .padding(.bottom, 10)
        .frame(width: 30, height: 30)

        Spacer()
    }
}

private var titleView: some View {
    VStack(spacing: 10) {
        Text(gameDefinitionModel?.name ??
StringConstants.Gamification.scratchCardTitleLabel)
            .font(.gameHeaderTitle)
        Text(gameDefinitionModel?.description ??
StringConstants.Gamification.scratchCardSubTitleLabel)
            .font(.gameHeaderSubTitle)
    }
    .padding(30)
}

private var loadingView: some View {
    Text("Loading...")
        .foregroundColor(.white)
        .font(.system(size: 24))
        .fontWeight(.bold)
}

```

```

        .frame(width: cardSize.width, height: cardSize.height)
        .background(Color.theme.accent)
    }

    private var rewardText: some View {
        Text(playGameViewModel.playedGameRewards?.first?.name ?? "--")
            .font(.largeTitle)
            .bold()
            .foregroundColor(.white)
            .frame(width: cardSize.width, height: cardSize.height)
            .background(Color.theme.accent)
            .zIndex(finishedScratching ? 10 : 0)
    }

    private var scratchCardWrapperText: some View {
        Text(String(repeating: (StringConstants.Gamification.scratchCardLabel), count:
70))
            .font(Font.scratchText)
            .multilineTextAlignment(.center)
            .foregroundColor(Color.theme.scratchCardText)
            .rotationEffect(Angle(degrees: -45))
            .mask {
                Rectangle()
                    .frame(width: cardSize.width, height: cardSize.height)
                    .cornerRadius(10)
                    .opacity(finishedScratching ? 0 : 1)
            }
            .opacity(finishedScratching ? 0 : 1)
    }

    private var scratchCardContentView: some View {
        ZStack {
            // Purple background with postage stamp border
            DottedBorderRectangle(width: backgroundSize.width,
                height: backgroundSize.height,
                color: Color.theme.accent)

            // Grey scratch card
            Rectangle()
                .fill(Color.theme.scratchCardBackground)
                .frame(width: cardSize.width, height: cardSize.height)
                .cornerRadius(10)
                .opacity(finishedScratching ? 0 : 1)

            // Text overlay
            scratchCardWrapperText
        }
    }

    private var scratchCardGame: some View {
        ScratchCardGame(cursorSize: 30, cardSize: cardSize, gameModel: gameDefinitionModel,
onFinish: $finishedScratching) {
            scratchCardContentView
        } overlayView: {

```



```

        // Reward text
        switch playGameViewModel.state {
        case .loaded:
            rewardText
        default:
            loadingView
        }
    }
    .animation(.linear(duration: 0.5), value: playGameViewModel.state)
    .environmentObject(playGameViewModel)
}

private var descriptionView: some View {
    VStack(spacing: 20) {
        Text((StringConstants.Gamification.scratchCardBodyLabel))
            .font(.gameDescText)
            .multilineTextAlignment(.center)
            .frame(width: 258)
    }
}
}

```

For example, to change the angle of the scratch card wrapper text inside the gray box, open the `LoyaltyMobileSDK-iOS/SampleApps/MyNTORewards/MyNTORewards/GameZone/Views/ScrachCard/ScratchCardView.swift` file in Xcode, and replace `rotationEffect (Angle (degrees: -45))` with `rotationEffect (Angle (degrees: -25))`. To change the number of times the text appears inside the gray box, replace `Text (String (repeating: (StringConstants.Gamification.scratchCardLabel), count: 70))` with `Text (String (repeating: (StringConstants.Gamification.scratchCardLabel), count: 50))`.

```

private var scratchCardWrapperText: some View {
    Text (String (repeating: (StringConstants.Gamification.scratchCardLabel), count:
50))
        .font (Font.scratchText)
        .multilineTextAlignment (.center)
        .foregroundColor (Color.theme.scratchCardText)
        .rotationEffect (Angle (degrees: -25))
        .mask {
            Rectangle ()
                .frame (width: cardSize.width, height: cardSize.height)
                .cornerRadius (10)
                .opacity (finishedScratching ? 0 : 1)
        }
        .opacity (finishedScratching ? 0 : 1)
}
}

```

## Gamification Mobile SDK for Android

Clone and import the Gamification Mobile Android SDK to integrate the gamification feature in a mobile app.

### EDITIONS

Available in: all editions that have Loyalty Management enabled

### [Clone and Import Gamification Mobile SDK for Android](#)

Clone the GitHub repository to copy the code base and revision history of the repository to your local computer. After you clone the mobile SDK for Android repository, import Gamification Mobile Android SDK in Android Studio to automatically download and manage external dependencies.

### [Integrate Gamification Mobile Android SDK into Loyalty Management Mobile Android SDK](#)

To integrate the Gamification Mobile Android SDK into the existing Loyalty Management Mobile Android SDK, add it as an external dependency.

### [Gamification Mobile SDK Android Classes](#)

The Android classes organize the Gamification Mobile SDK's functionalities and data structures. When developing mobile apps using the SDK, these classes provide components, services, or features with which you can interact.

### [Gamification Mobile Android SDK API Reference](#)


The Gamification Mobile Android SDK interacts with the Salesforce Gamification APIs, and retrieves games assigned to a loyalty member, and the rewards won by the member after playing the game.


### [Game Zone UI Customization for Android](#)

After you build the Gamification Mobile SDK for Android, you can customize the game zone UI components.

## Clone and Import Gamification Mobile SDK for Android

Clone the GitHub repository to copy the code base and revision history of the repository to your local computer. After you clone the mobile SDK for Android repository, import Gamification Mobile Android SDK in Android Studio to automatically download and manage external dependencies.

1. Go to the [Gamification Mobile SDK Android repository](#).
2. To copy the repository URL to your clipboard, click **Code**, and then click .
3. Open Terminal, and go to the directory where you want to clone the repository.
4. Enter `git clone https://github.com/loyaltysampleapp/GamificationMobileSDK-Android`.
5. Press Enter and wait for the repository to clone.
6. Unzip the `GamificationMobileSDK-Android` package that you downloaded.
7. Open Android Studio.
8. Go to **File > New > Import Project**, and then select the `GamificationMobileSDK-Android` folder.
9. Click **Open**.
10. Open the SDK files and ensure that the files include the appropriate SDK packages.

 **Note:** If the sync fails, disconnect the VPN to ensure there's no blocker when downloading the dependencies, and then resync.

You can now use the classes and interfaces in the `GamificationMobileSDK-Android` module to manage and customize the gamification features in your application.

### EDITIONS

Available in: all editions that have Loyalty Management enabled

# Integrate Gamification Mobile Android SDK into Loyalty Management Mobile Android SDK

To integrate the Gamification Mobile Android SDK into the existing Loyalty Management Mobile Android SDK, add it as an external dependency.

1. Open Terminal, and go to the `LoyaltyMobileSDK-Android` folder.

2. Run these commands:

```
git submodule add
git@github.com:loyaltysampleapp/GamificationMobileSDK-Android.git
git submodule sync
git submodule update
cd GamificationMobileSDK-Android
git submodule update --init --recursive --remote
```

3. Open the `LoyaltyMobileSDK-Android` project in Android Studio.

4. Click **Open**.

5. Open the `SampleApps/MyNTORewards/build.gradle` file, and under Dependencies block, to the first line of Dependencies, add:

```
implementation(project(":GamificationMobileSDK-Android:GamificationMobileSDK"))
```

6. Open the `SampleApps/MyNTORewards/connectedapp.properties.sample` file and modify the properties of the connected app as per your org.

7. Click **Sync Now**.

8. Open the SDK files and ensure that the files include the appropriate SDK packages.



**Note:** If the sync fails, disconnect the VPN to ensure there's no blocker when downloading the dependencies, and then resync.

9. To add the gamification SDK package, include this code snippet in your code.

```
`import com.salesforce.gamification.api.*
import com.salesforce.gamification.model.*
import com.salesforce.gamification.repository.*
import com.salesforce.gamification.utilities.*`
```

You can now use the classes and interfaces in the `GamificationMobileSDK-Android` module to extend and manage the gamification features in the `MyNTORewards` sample app.

## Gamification Mobile SDK Android Classes

The Android classes organize the Gamification Mobile SDK's functionalities and data structures. When developing mobile apps using the SDK, these classes provide components, services, or features with which you can interact.

### EDITIONS

Available in: all editions that have Loyalty Management enabled

### EDITIONS

Available in: all editions that have Loyalty Management enabled

[GamificationRemoteRepository](#)

The `GamificationRemoteRepository` class invokes and manages gamification-related REST API requests. Use this class to interact with the Salesforce gamification APIs and retrieve member games, and play a game.

[GameAPIClient](#)

The `GameAPIClient` class creates a retrofit instance to invoke REST APIs.

[GameAuthenticator](#)

The `GameAuthenticator` interface defines the necessary methods for handling access tokens in the Salesforce API.

## GamificationRemoteRepository

The `GamificationRemoteRepository` class invokes and manages gamification-related REST API requests. Use this class to interact with the Salesforce gamification APIs and retrieve member games, and play a game.

The Gamification Mobile Android SDK requires a valid access token to interact with the Salesforce gamification APIs. To provide a valid access token, implement the `GameAuthenticator` interface in your `GamificationRemoteRepository` class at the time of initialization. This class manages authentication using the provided instance of `GameAuthenticator`.

Use this constructor to initialize the `GamificationRemoteRepository` class.

```
class GamificationRemoteRepository constructor(
    auth: GameAuthenticator,
    instanceUrl: String,
    gameClient: NetworkClient
)
```

Parameter	Type	Description	Required or Optional
<code>auth</code>	<code>GameAuthenticator</code>	An instance of the <code>GameAuthenticator</code> interface.	Required
<code>instanceUrl</code>	<code>String</code>	The URL of your Salesforce org.	Required
<code>gameClient</code>	<code>NetworkClient</code>	An instance of the <code>GameAPIClient</code> class.	Required

## Methods

### getGames

Retrieve games for the specified participant ID.

```
final Result<Games> getGames(String participantId, String gameParticipantRewardId, Boolean mockResponse)
```

Parameter	Type	Description	Required or Optional
<code>participantId</code>	<code>String</code>	The ID of a loyalty program member, an account, or a	Required

Parameter	Type	Description	Required or Optional
		contact whose eligible games are to be retrieved.	
gameParticipantRewardId	String	The ID of the participant's game reward associated with the game whose details are to be fetched.	Optional
mockResponse	Boolean	Indicates whether to include a sample response ( <code>true</code> ) or not ( <code>false</code> ).	Optional

## getGameReward

Retrieve the details of the reward that's provided to a participant after the game play.

```
final Result<GameRewardResponse> getGameReward(String gameParticipantRewardId, Boolean mockResponse)
```

Parameter	Type	Description	Required or Optional
gameParticipantRewardId	String	The ID of the participant's game reward won by the participant whose details are to be fetched.	Required
mockResponse	Boolean	Indicates whether to include a sample response ( <code>true</code> ) or not ( <code>false</code> ).	Optional

## GameAPIClient

The `GameAPIClient` class creates a retrofit instance to invoke REST APIs.

This class extends the `NetworkClient` abstract class.

Use this constructor to initialize the `GameAPIClient` class.

```
class GameAPIClient constructor(auth: GameAuthenticator, instanceUrl: String) : NetworkClient(auth, instanceUrl)
```

Parameter	Type	Description	Required or Optional
auth	GameAuthenticator	An instance of the <code>GameAuthenticator</code> interface.	Required
instanceUrl	String	The URL of your Salesforce org.	Required

## Methods

### getOkHttpClientBuilder

Retrieve an instance of `OkHttpClient`.

```
final OkHttpClient getOkHttpClientBuilder()
```

### getNetworkClient

Override the `getNetworkClient` method of the `NetworkClient` abstract class to get an instance of `GameAPIInterface`.

```
final GameAPIInterface getNetworkClient()
```

## GameAuthenticator

The `GameAuthenticator` interface defines the necessary methods for handling access tokens in the Salesforce API.

## Methods

### getAccessToken()

Returns a valid access token as a string if available, otherwise null.

```
final String getAccessToken()
```

### grantAccessToken

Returns a refreshed or newly granted access token as a string.

```
final String grantAccessToken()
```

## Gamification Mobile Android SDK API Reference

The Gamification Mobile Android SDK interacts with the Salesforce Gamification APIs, and retrieves games assigned to a loyalty member, and the rewards won by the member after playing the game.

Interact with the Gamification APIs, including:

Resource	Description
<a href="#">/game/participant/{participantId}/games</a> on page 344	Get games assigned to a participant.
<a href="#">/game/participant/{participantId}/game-reward</a> on page 344	Get details of the reward won after the member plays the game.

### EDITIONS

Available in: all editions that have Loyalty Management enabled

## Game Zone UI Customization for Android

After you build the Gamification Mobile SDK for Android, you can customize the game zone UI components.

The game zone UI components are available in the

`/LoyaltyMobileSDK-Android/SampleApps/MyNCRewards/src/main/java/com/salesforce/loyalty/mobile/myntorewards/views/gamezone` package.

## Spinner Customization

Customize the spinner components, such as the wheel size, frame width and color, and more in the `SpinnerConfiguration.kt` file. Here are all the parameters you can modify.

```
class SpinnerConfiguration {

    companion object {
        //preferences
        val WHEEL_SIZE = 280.dp
        val WHEEL_FRAME_WIDTH = 4.dp
        val WHEEL_FRAME_COLOUR = SpinnerFrameColour
        val WHEEL_BORDER_WIDTH = 12.dp
        val WHEEL_SEGMENT_BACKGROUND = Color.White //same colour will apply on border
        const val START_DEGREE = 0f
        //spinner default loading position degree 30f represents when spinner will load
        it will be at 30 degree
        const val ROTATION_PER_SECOND = 3f
        const val PADDING_ANGLE_BETWEEN_WHEEL_SEGMENT = 2f //padding between two segments
        of wheel
        const val ROTATION_DURATION = 3000 // duration of wheel rotate after the API
        responds
        const val INITIAL_ROTATION_DURATION = 20000 // duration of wheel rotate till API
        responds
        const val WHEEL_TEXT_INITIAL_ROTATION = 0f // we can configure the orientation of
        text
        //based on this angle,we can make wheel content straight or with angle inside the
        wheel
    }
}
```

For example, to change the spinner frame color to blue, open the

`/LoyaltyMobileSDK-Android/SampleApps/MyNCRewards/src/main/java/com/salesforce/loyalty/mobile/myntorewards/views/gamezone/spinner/SpinnerConfiguration.kt` file in Android Studio, and change the value of `val WHEEL_FRAME_COLOUR` from `SpinnerFrameColour` to `Color.Blue`.

To increase the padding between two segments of the wheel, change the value of `const val`

`PADDING_ANGLE_BETWEEN_WHEEL_SEGMENT` from `2f` to `4f`.

## Scratch Card Customization

You can customize the scratch card components, such as the box and canvas in the `ScratchCanvasView.kt` file.

For example, to change the scratch card background color to white, open the

`/LoyaltyMobileSDK-Android/SampleApps/MyNCRewards/src/main/java/com/salesforce/loyalty/mobile/myntorewards/views/gamezone/spinner/ScratchCanvasView.kt` file in Android Studio, and inside the `CanvasForScratching` method, replace `Box(modifier = modifier.background(VibrantPurple40).testTag(TEST_TAG_SCRATCH_CARD))` with `Box(modifier = modifier.background(Color.White).testTag(TEST_TAG_SCRATCH_CARD))`.

# CHAPTER 13 Data Processing Engine, Batch Management, and Monitor Workflow Services

## In this chapter ...

- [Data Model](#)
- [Common Tooling API Object](#)
- [Common Platform Event](#)
- [Common Business APIs](#)
- [Data Processing Engine](#)
- [Batch Management](#)
- [Monitor Workflow Services](#)

Data Processing Engine and Batch Management help automate your business processes. Use objects, APIs, Platform Events, and invocable actions to define, run, and review Data Processing Engine definitions and Batch Management jobs.

Here's how both these features can automate your business processes:

- **Data Processing Engine:** Transform data that's available in your Salesforce org and write back the transformation results as new or updated records. You can transform the data for standard and custom objects.
- **Batch Management:** Automate the processing of records in scheduled flows. You can process a high volume of standard and custom object records.

Once a Data Processing Engine definition is run or a Batch Management job is run, you can view the progress of the run and the results of the run using Monitor Workflow Services.

## EDITIONS

Available in: Lightning Experience

Available in: Data Processing Engine is available with Enterprise, Unlimited, and Performance Editions with the Financial Services Cloud, Loyalty Management, Manufacturing Cloud, Rebate Management, Accounting Subledger, or Provider Search in Health Cloud.

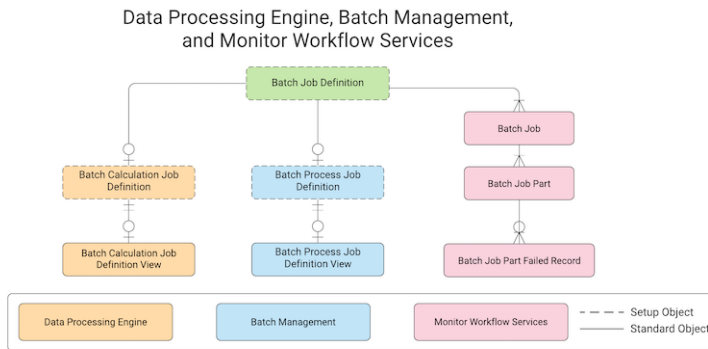
Available in: Batch Management is available with Enterprise, Unlimited, and Performance Editions with the Loyalty Management, Manufacturing Cloud, Rebate Management, or Accounting Subledger



# Data Model

Data Processing Engine, Batch Management, and Monitor Workflow Services share a data model. Let's learn about the objects and relationships in this shared data model.

Here's the data model:



## EDITIONS

Available in: Lightning Experience

Available in: Data Processing Engine is available with Enterprise, Unlimited, and Performance Editions with the Financial Services Cloud, Loyalty Management, Manufacturing Cloud, Rebate Management, Accounting Subledger, or Provider Search in Health Cloud.

Available in: Batch Management is available with Enterprise, Unlimited, and Performance Editions with the Loyalty Management, Manufacturing Cloud, Rebate Management, or Accounting Subledger

# Common Tooling API Object

BatchJobDefinition is a common Tooling API object that is shared between Data Processing Engine and Batch Management.


Tooling API objects let you interact with metadata for declarative development. For example, you can create your own version of Setup.

## [BatchJobDefinition](#)

Represents the definition of a batch job. This object is available in API version 51.0 and later.

## BatchJobDefinition

Represents the definition of a batch job. This object is available in API version 51.0 and later.

 **Note:** 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

`describeSObjects()`, `query()`, `retrieve()`

## Supported REST API Methods

GET, HEAD, Query

## Fields

Field	Details
Description	<p><b>Type</b> textarea</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The description of the batch job definition.</p>
DeveloperName	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Sort</p> <p><b>Description</b> The developer name of the batch job.</p>
Language	<p><b>Type</b> picklist</p>

### EDITIONS

Available in: Lightning Experience

Available in: Data Processing Engine is available with Enterprise, Unlimited, and Performance Editions with the Financial Services Cloud, Loyalty Management, Manufacturing Cloud, Rebate Management, Accounting Subledger, or Provider Search in Health Cloud.

Available in: Batch Management is available with Enterprise, Unlimited, and Performance Editions with the Loyalty Management, Manufacturing Cloud, Rebate Management, or Accounting Subledger

**Field**

**Details**

**Properties**

Filter, Group, Restricted picklist, Sort

**Description**

The language in which the batch job is created.

Possible values are:

- af—Afrikaans
- am—Amharic
- ar—Arabic
- ar\_AE—Arabic (United Arab Emirates)
- ar\_BH—Arabic (Bahrain)
- ar\_DZ—Arabic (Algeria)
- ar\_EG—Arabic (Egypt)
- ar\_IQ—Arabic (Iraq)
- ar\_JO—Arabic (Jordan)
- ar\_KW—Arabic (Kuwait)
- ar\_LB—Arabic (Lebanon)
- ar\_LY—Arabic (Libya)
- ar\_MA—Arabic (Morocco)
- ar\_OM—Arabic (Oman)
- ar\_QA—Arabic (Qatar)
- ar\_SA—Arabic (Saudi Arabia)
- ar\_SD—Arabic (Sudan)
- ar\_SY—Arabic (Syria)
- ar\_TN—Arabic (Tunisia)
- ar\_YE—Arabic (Yemen)
- bg—Bulgarian
- bn—Bengali
- bs—Bosnian
- ca—Catalan
- cs—Czech
- cy—Welsh
- da—Danish
- de—German
- de\_AT—German (Austria)
- de\_BE—German (Belgium)
- de\_CH—German (Switzerland)
- de\_LU—German (Luxembourg)
- el—Greek

Field	Details
	<ul style="list-style-type: none"><li>• en_AU—English (Australian)</li><li>• en_CA—English (Canadian)</li><li>• en_GB—English (UK)</li><li>• en_HK—English (Hong Kong)</li><li>• en_IE—English (Ireland)</li><li>• en_IN—English (Indian)</li><li>• en_MY—English (Malaysian)</li><li>• en_NZ—English (New Zealand)</li><li>• en_PH—English (Phillipines)</li><li>• en_SG—English (Singapore)</li><li>• en_US—English</li><li>• en_ZA—English (South Africa)</li><li>• eo—Esperanto (Pseudo)</li><li>• es—Spanish</li><li>• es_AR—Spanish (Argentina)</li><li>• es_BO—Spanish (Bolivia)</li><li>• es_CL—Spanish (Chile)</li><li>• es_CO—Spanish (Colombia)</li><li>• es_CR—Spanish (Costa Rica)</li><li>• es_DO—Spanish (Dominican Republic)</li><li>• es_EC—Spanish (Ecuador)</li><li>• es_GT—Spanish (Guatemala)</li><li>• es_HN—Spanish (Honduras)</li><li>• es_MX—Spanish (Mexico)</li><li>• es_NI—Spanish (Nicaragua)</li><li>• es_PA—Spanish (Panama)</li><li>• es_PE—Spanish (Peru)</li><li>• es_PR—Spanish (Puerto Rico)</li><li>• es_PY—Spanish (Paraguay)</li><li>• es_SV—Spanish (El Salvador)</li><li>• es_US—Spanish (United States)</li><li>• es_UY—Spanish (Uruguay)</li><li>• es_VE—Spanish (Venezuela)</li><li>• et—Estonian</li><li>• eu—Basque</li><li>• fa—Farsi</li><li>• fi—Finnish</li><li>• fr—French</li></ul>

Field	Details
	<ul style="list-style-type: none"> <li>• fr_BE—French (Belgium)</li> <li>• fr_CA—French (Canadian)</li> <li>• fr_CH—French (Switzerland)</li> <li>• fr_LU—French (Luxembourg)</li> <li>• ga—Irish</li> <li>• gu—Gujarati</li> <li>• hi—Hindi</li> <li>• hr—Croatian</li> <li>• hu—Hungarian</li> <li>• hy—Armenian</li> <li>• in—Indonesian</li> <li>• is—Icelandic</li> <li>• it—Italian</li> <li>• it_CH—Italian (Switzerland)</li> <li>• iw—Hebrew</li> <li>• iw_EO—Esperanto RTL (Pseudo)</li> <li>• ja—Japanese</li> <li>• ka—Georgian</li> <li>• km—Khmer</li> <li>• kn—Kannada</li> <li>• ko—Korean</li> <li>• lb—Luxembourgish</li> <li>• lt—Lithuanian</li> <li>• lv—Latvian</li> <li>• mi—Te reo</li> <li>• mk—Macedonian</li> <li>• ml—Malayalam</li> <li>• mr—Marathi</li> <li>• ms—Malay</li> <li>• mt—Maltese</li> <li>• my—Burmese</li> <li>• nl_BE—Dutch (Belgium)</li> <li>• nl_NL—Dutch</li> <li>• no—Norwegian</li> <li>• pl—Polish</li> <li>• pt_BR—Portuguese (Brazil)</li> <li>• pt_PT—Portuguese (European)</li> <li>• rm—Romansh</li> </ul>

Field	Details
	<ul style="list-style-type: none"> <li>• ro—Romanian</li> <li>• ro_MD—Romanian (Moldova)</li> <li>• ru—Russian</li> <li>• sh—Serbian (Latin)</li> <li>• sh_ME—Montenegrin</li> <li>• sk—Slovak</li> <li>• sl—Slovene</li> <li>• sq—Albanian</li> <li>• sr—Serbian (Cyrillic)</li> <li>• sv—Swedish</li> <li>• sw—Swahili</li> <li>• ta—Tamil</li> <li>• te—Telugu</li> <li>• th—Thai</li> <li>• tl—Tagalog</li> <li>• tr—Turkish</li> <li>• uk—Ukrainian</li> <li>• ur—Urdu</li> <li>• vi—Vietnamese</li> <li>• xh—Xhosa</li> <li>• zh_CN—Chinese (Simplified)</li> <li>• zh_HK—Chinese (Hong Kong)</li> <li>• zh_SG—Chinese (Singapore)</li> <li>• zh_TW—Chinese (Traditional)</li> <li>• zu—Zulu</li> </ul>
MasterLabel	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Sort</p> <p><b>Description</b> The label of the batch job.</p>
ProcessGroup	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Sort</p>

Field	Details
	<p><b>Description</b> The group or team that's using the batch job. This field is only applicable to Batch Management jobs.</p>
Status	<p><b>Type</b> picklist</p> <p><b>Properties</b> Defaulted on create, Filter, Group, Restricted picklist, Sort</p> <p><b>Description</b> The status of the batch job.</p> <p>Possible values are:</p> <ul style="list-style-type: none"><li>• Active</li><li>• Inactive</li></ul>
Type	<p><b>Type</b> picklist</p> <p><b>Properties</b> Defaulted on create, Filter, Group, Restricted picklist, Sort</p> <p><b>Description</b> The type of batch job.</p> <p>Possible values are:</p> <ul style="list-style-type: none"><li>• Calc—Data Processing Engine</li><li>• DecisionTableRefresh</li><li>• Flow</li></ul>

# Common Platform Event

Batch Management jobs and Data Processing Engine definitions are run using invocable actions in Flows. Use the BatchJobStatusChanged event to notify subscribers after a Batch Management job or a Data Processing Engine definition is processed in a flow.

## BatchJobStatusChangedEvent

Notifies subscribers of when a batch job is completed in a flow. This object is available in API version 51.0 and later.

## BatchJobStatusChangedEvent

Notifies subscribers of when a batch job is completed in a flow. This object is available in API version 51.0 and later.

## Supported Calls

describeSObjects ()

## Supported Subscribers

Subscriber	Supported?
Flows	✓

## Fields

Field	Details
BatchJob	<p><b>Type</b> string</p> <p><b>Description</b> The unique identifier of the batch job.</p>
BatchJobDefinition	<p><b>Type</b> string</p> <p><b>Properties</b> Nillable</p> <p><b>Description</b> The unique identifier of the batch job's definition.</p>
EndDateTime	<p><b>Type</b> dateTime</p>

### EDITIONS

Available in: Lightning Experience

Available in: Data Processing Engine is available with Enterprise, Unlimited, and Performance Editions with the Financial Services Cloud, Loyalty Management, Manufacturing Cloud, Rebate Management, Accounting Subledger, or Provider Search in Health Cloud.

Available in: Batch Management is available with Enterprise, Unlimited, and Performance Editions with the Loyalty Management, Manufacturing Cloud, Rebate Management, or Accounting Subledger



Field	Details
	<p><b>Properties</b> Nillable</p> <p><b>Description</b> The timestamp for when the batch job execution is complete.</p>
EventUuid	<p><b>Type</b> string</p> <p><b>Properties</b> Nillable</p> <p><b>Description</b> A universally unique identifier (UUID) that identifies a platform event message. This field is available in API version 52.0 and later.</p>
ReplayId	<p><b>Type</b> string</p> <p><b>Properties</b> Nillable</p> <p><b>Description</b> 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.</p>
StartDateTime	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Nillable</p> <p><b>Description</b> The timestamp for when the batch job execution is started.</p>
Status	<p><b>Type</b> picklist</p> <p><b>Properties</b> Restricted picklist</p> <p><b>Description</b> The status of the batch job. Possible values are:</p> <ul style="list-style-type: none"> <li>• Canceled—Canceled</li> <li>• Failure</li> <li>• Success</li> </ul>

## Common Business APIs

---

Common Business APIs are RESTful APIs that are sometimes available as Apex classes and methods.

### [REST Reference](#)

You can access Common Business APIs using REST endpoints. These REST APIs follow similar conventions as Connect REST APIs.

## REST Reference

You can access Common Business APIs using REST endpoints. These REST APIs follow similar conventions as Connect REST APIs.

To understand the architecture, authentication, rate limits, and how the requests and responses work, see [Connect REST API Developer Guide](#).

### [Resources](#)

Here's a list of Common Business API resources.

### [Response Bodies](#)

Here's a list of Common Business API response bodies.

SEE ALSO:

[Connect REST API Developer Guide](#)

## Resources

Here's a list of Common Business API resources.

### [Batch Job Cancel](#)

Cancel a batch job of type data processing engine (calc job) and batch management. A batch job with only the status Submitted or In Progress can be canceled.

## Batch Job Cancel

Cancel a batch job of type data processing engine (calc job) and batch management. A batch job with only the status Submitted or In Progress can be canceled.

### Special Access Rules

To use this resource, the following permissions are required:

- Your org must have the Batch Management and Data Processing Engine licenses
- Users in your org require System Administration profile

### Resource

```
/connect/batch-job/batchJobId/cancel-job
```

### Resource example

```
/connect/batch-job/0mdxx00000000fxAAA/cancel-job
```

**Available version**

52.0

**Requires Chatter**

No

**HTTP methods**

POST

**Note:** POST doesn't take request parameters or a request body.**Response body for POST**

Returns HTTP 201 on success.

See [Batch Job Cancel Output](#) for HTTP code descriptions that are unique to this resource in case of failure of the batch job cancel request.

## Response Bodies

Here's a list of Common Business API response bodies.

[Batch Job Cancel Output](#)

Output representation of the batch job cancel request.

### Batch Job Cancel Output

Output representation of the batch job cancel request.

Property Name	Type	Description	Filter Group and Version	Available Version
message	String	Details about why the batch job cancel request failed.	Small, 52.0	52.0

This table lists HTTP response code descriptions that are unique to this resource.

HTTP Response Code	Error Code	Description
400	INVALID_STATUS	We can't cancel the batch job that doesn't have an active run. Specify the ID of a batch job with an active run and try again.
400	INVALID_STATUS	We can't cancel the batch job that is already canceled or completed. Specify the ID of a valid batch job with the status InProgress or Submitted and try again.
400	DELETE_FAILED	We can't cancel the batch job of which the results are already being written back.
400	DELETE_FAILED	We can't cancel the batch job because of an error in processing your org's data. Run the Data Processing Engine definition and try again.

HTTP Response Code	Error Code	Description
403	FORBIDDEN	You don't have the permission to cancel a batch job. Ask your Salesforce admin for help.
404	RESOURCE_NOT_FOUND	Specify the ID of a valid batch job and try again.
500	INTERNAL_SERVER_ERROR	Something went wrong when we tried to cancel the batch job. Try again or ask your Salesforce admin for help.

## Data Processing Engine

Transform data that's available in your Salesforce org and write back the transformation results as new or updated records. You can transform the data for standard and custom objects. Data Processing Engine consists of a Tooling API object, a standard object, a Metadata API, and an invocable action. You can use these to view, create, edit, and run Data Processing Engine definitions.

### [Data Processing Engine Tooling API Objects](#)

Data Processing Engine consists of one Tooling API object, BatchCalcJobDefinition. Use this object to create and edit a Data Processing Engine definition.

### [Data Processing Engine Standard Object](#)

Data Processing Engine contains one standard object, BatchCalcJobDefinitionView. Use this object to view all the Data Processing Engine definitions available in your Salesforce org, including file-based definitions.

### [Data Processing Engine Metadata API](#)

Use a Metadata API to create, update, and activate Data Processing Engine definitions.

### [Data Processing Engine Invocable Actions](#)

Run an active Data Processing Engine definition. For more information on custom invocable actions, see [REST API Developer Guide](#) and [Actions Developer Guide](#).

### EDITIONS

Available in: Lightning Experience

Available in: Data Processing Engine is available with Enterprise, Unlimited, and Performance Editions with the Financial Services Cloud, Loyalty Management, Manufacturing Cloud, Rebate Management, Accounting Subledger, or Provider Search in Health Cloud.

## Data Processing Engine Tooling API Objects

Data Processing Engine consists of one Tooling API object, BatchCalcJobDefinition. Use this object to create and edit a Data Processing Engine definition.


Tooling API objects let you interact with metadata for declarative development. For example, you can create your own version of Setup.

### [BatchCalcJobDefinition](#)

Represents a Data Processing Engine (DPE) definition. This object is available in API version 51.0 and later.

### BatchCalcJobDefinition

Represents a Data Processing Engine (DPE) definition. This object is available in API version 51.0 and later.

 **Note:** 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

`describeSObjects()`, `query()`, `retrieve()`

### Supported REST API Methods

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

### Fields

Field	Details
BatchJobDefinitionId	<p><b>Type</b> reference</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The unique identifier of the associated batch job definition. This is a relationship field.</p> <p><b>Relationship Name</b> BatchJobDefinition</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> BatchJobDefinition</p>

### EDITIONS

Available in: Lightning Experience

Available in: Data Processing Engine is available with Enterprise, Unlimited, and Performance Editions with the Financial Services Cloud, Loyalty Management, Manufacturing Cloud, Rebate Management, Accounting Subledger, or Provider Search in Health Cloud.

Field	Details
DefinitionRunMode	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p><b>Description</b> Possible values are:</p> <ul style="list-style-type: none"> <li>• Batch</li> <li>• OnDemand—On Demand</li> </ul> <p>The default value is <code>Batch</code>.</p>
DeveloperName	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Sort</p> <p><b>Description</b> The developer name of the Data Processing Engine definition.</p>
ExecutionPlatformType	<p><b>Type</b> Picklist</p> <p><b>Properties</b> <b>Filter, Group, Restricted picklist, Sort, Not-Nillable</b></p> <p><b>Description</b> Specifies the platform that's used to run the Data Processing Engine definition. Possible values:</p> <ul style="list-style-type: none"> <li>• CRMA—CrmAnalytics</li> <li>• CDP—DataCloudForMarketing</li> <li>• CORE—This value is reserved for internal use.</li> </ul>
FullName	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Group, Nillable</p> <p><b>Description</b> The name of the Data Processing Engine definition.</p> <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
IsTemplate	<p><b>Type</b> boolean</p>

Field	Details
	<p><b>Properties</b> Defaulted on create, Filter, Group, Sort</p> <p><b>Description</b> Indicates whether this is a template Data Processing Engine definition.</p>
Language	<p><b>Type</b> picklist</p> <p><b>Properties</b> Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p><b>Description</b> The language in which this Data Processing Engine definition is created. Possible values are:</p> <ul style="list-style-type: none"> <li>• da—Danish</li> <li>• de—German</li> <li>• en_US—English</li> <li>• es—Spanish</li> <li>• es_MX—Spanish (Mexico)</li> <li>• fi—Finnish</li> <li>• fr—French</li> <li>• it—Italian</li> <li>• ja—Japanese</li> <li>• ko—Korean</li> <li>• nl_NL—Dutch</li> <li>• no—Norwegian</li> <li>• pt_BR—Portuguese (Brazil)</li> <li>• ru—Russian</li> <li>• sv—Swedish</li> <li>• th—Thai</li> <li>• zh_CN—Chinese (Simplified)</li> <li>• zh_TW—Chinese (Traditional)</li> </ul>
ManageableState	<p><b>Type</b> ManageableState enumerated list</p> <p><b>Properties</b> Filter, Group, Nillable, Restricted picklist, Sort</p> <p><b>Description</b> Indicates the manageable state of the specified component that is contained in a package:</p> <ul style="list-style-type: none"> <li>• beta</li> <li>• deleted</li> </ul>

Field	Details
	<ul style="list-style-type: none"><li>• deprecated</li><li>• deprecatedEditable</li><li>• installed</li><li>• installedEditable</li><li>• released</li><li>• unmanaged</li></ul>
MasterLabel	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Sort</p> <p><b>Description</b> Label of the Data Processing Engine definition.</p>
Metadata	<p><b>Type</b> complexvalue</p> <p><b>Properties</b> Create, Nillable, Update</p> <p><b>Description</b> Data Processing Engine definition's metadata.  Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
NamespacePrefix	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> 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 <b><i>namespacePrefix__componentName</i></b> notation.  The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"><li>• 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.</li><li>• In organizations that aren't Developer Edition organizations, <code>NamespacePrefix</code> is only set for objects that are part of an installed managed package. There's no namespace prefix for all other objects.</li></ul>



Field	Details
ProcessType	<p><b>Type</b> picklist</p> <p><b>Properties</b> Filter, Group, Restricted picklist, Sort</p> <p><b>Description</b> Required. The process type for which the Data Processing Engine definition is created. Possible values are:</p> <ul style="list-style-type: none"> <li>• AccountingPeriodClosure—Legal Entity Accounting Period Closure</li> <li>• AccountingSubledger—This value is reserved for internal use.</li> <li>• ActionableList</li> <li>• AdvancedAccountForecast</li> <li>• BenefitManagement—Available in API version 61.0 and later.</li> <li>• BillingSchedulesforInvoiceGeneration—Billing Schedules for Invoice Generation</li> <li>• ChannelInventoryManagement—Available in API version 63.0 and later.</li> <li>• CriteriaBsdSearchAndFilter - Criteria-Based Search And Filter</li> <li>• CSVCoreUpload—This value is reserved for internal use.</li> <li>• DataProcessingEngine - Standard</li> <li>• DecisionMatrixDataUpload - This value is available only if you have Business Rules Engine enabled.</li> <li>• Decisiontable—Decision table activation</li> <li>• EmployeeService—Available in API version 63.0 and later.</li> <li>• FinancialSummaryRollup—Available in API version 62.0 and later.</li> <li>• FSHierarchyRollUp</li> <li>• FundraisingRollups—Available in API version 63.0 and later.</li> <li>• InvoiceGeneration—Available in API version 62.0 and later.</li> <li>• LegalEntityAccountingPeriodClosureAdvanced—Available in API version 63.0 and later.</li> <li>• LifeSciencesCommercialTerritoryAlignment—Available in API version 63.0 and later.</li> <li>• Loyalty</li> <li>• LoyaltyPartnerManagement</li> <li>• LoyaltyPointsAggregation</li> <li>• NetZero</li> <li>• NextGenForecasting—Available in API version 62.0 and later.</li> <li>• PnmRosterFileUpload—Available in API version 62.0 and later.</li> <li>• PriceProtection—Available in API version 62.0 and later.</li> <li>• ProductCatalogManagement—Available in API version 62.0 and later.</li> </ul>

Field	Details
	<ul style="list-style-type: none"> <li>• <code>ProgramBasedBusiness</code></li> <li>• <code>ProgramManagementRollups</code></li> <li>• <code>ProviderSearch</code>—This value is reserved for internal use.</li> <li>• <code>Rebates</code></li> <li>• <code>Recruitment</code>—Available in API version 62.0 and later.</li> <li>• <code>RevenueTransactionManagement</code>—Available in API version 63.0 and later.</li> <li>• <code>SalesAgreement</code>—Available in API version 63.0 and later.</li> <li>• <code>TestProcessType</code>—Available in API version 62.0 and later.</li> <li>• <code>UsageManagement</code>—Available in API version 62.0 and later.</li> </ul> <p>When Data Processing Engine is enabled for a Salesforce org, the default value is 'Standard'. Other process types may be available to you depending on your industry solution and permission sets.</p>

## Data Processing Engine Standard Object


Data Processing Engine contains one standard object, `BatchCalcJobDefinitionView`. Use this object to view all the Data Processing Engine definitions available in your Salesforce org, including file-based definitions.

### [BatchCalcJobDefinitionView](#)

Represents the details of a Data Processing Engine definition. The definition can also be a file-based definition that is available in your Salesforce org. This object is available in API version 51.0 and later.

### BatchCalcJobDefinitionView

Represents the details of a Data Processing Engine definition. The definition can also be a file-based definition that is available in your Salesforce org. This object is available in API version 51.0 and later.

 **Note:** 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

`describeSObjects()`, `query()`

### Fields

Field	Details
<code>DataSpaceApiName</code>	<p><b>Type</b> string</p>

### EDITIONS

Available in: Lightning Experience

Available in: Data Processing Engine is available with Enterprise, Unlimited, and Performance Editions with the Financial Services Cloud, Loyalty Management, Manufacturing Cloud, Rebate Management, Accounting Subledger, or Provider Search in Health Cloud.

Field	Details
	<p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> Stores the data space API name from Data Cloud.</p>
Description	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The description of a Data Processing Engine definition.</p>
DefinitionRunMode	<p><b>Type</b> picklist</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p><b>Description</b> Possible values are:</p> <ul style="list-style-type: none"> <li>• Batch</li> <li>• OnDemand—On Demand</li> </ul> <p>The default value is <code>Batch</code>.</p>
DurableId	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The unique identifier for the field. Always retrieve this value before using it, as the value isn't guaranteed to stay the same from one release to the next. Simplify queries by using this field instead of making multiple queries.</p>
ExecutionPlatformType	<p><b>Type</b> Picklist</p> <p><b>Properties</b> <b>Filter, Group, Restricted picklist, Sort, Not-Nillable</b></p> <p><b>Description</b> Specifies the platform that's used to run the Data Processing Engine definition. Possible values:</p> <ul style="list-style-type: none"> <li>• CRMA: CrmAnalytics</li> <li>• CDP: DataCloudForMarketing</li> </ul>

Field	Details
InstalledPackageName	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The name of the package used to add the definition to the org.</p>
IsActive	<p><b>Type</b> boolean</p> <p><b>Properties</b> Defaulted on create, Filter, Group, Sort</p> <p><b>Description</b> Indicates whether the definition is active.</p>
IsTemplate	<p><b>Type</b> boolean</p> <p><b>Properties</b> Defaulted on create, Filter, Group, Sort</p> <p><b>Description</b> Indicates whether the definition is a template. You can make a copy of a template definition and update it based on your requirements.</p>
LastModifiedBy	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The unique identifier of the user who modified the definition last.</p>
ManageableState	<p><b>Type</b> picklist</p> <p><b>Properties</b> Filter, Group, Nillable, Restricted picklist, Sort</p> <p><b>Description</b> Indicates the manageable state of the definition if it's contained in a package. Possible values are:</p> <ul style="list-style-type: none"> <li>• beta—Managed-Beta</li> <li>• deleted—Managed-Proposed-Deleted</li> <li>• deprecated—Managed-Proposed-Deprecated</li> <li>• deprecatedEditable—SecondGen-Installed-Deprecated</li> <li>• installed—Managed-Installed</li> </ul>

Field	Details
	<ul style="list-style-type: none"> <li>• <code>installedEditable</code>—SecondGen-Installed-Editable</li> <li>• <code>released</code>—Managed-Released</li> <li>• <code>unmanaged</code>—Unmanaged</li> </ul>
MasterLabel	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The label of the Data Processing Engine definition.</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The name of the Data Processing Engine definition.</p>
NamespacePrefix	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> 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 <code>namespacePrefix__componentName</code> notation.</p> <p>The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> <li>• 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.</li> <li>• In organizations that aren't 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.</li> </ul>
ProcessType	<p><b>Type</b> picklist</p> <p><b>Properties</b> Filter, Group, Nillable, Restricted picklist, Sort</p>

**Field**

**Details**

**Description**

The process type of the definition.

Possible values are:

- `AccountingPeriodClosure`—Legal Entity Accounting Period Closure—Available in API version 62.0 and later.
- `ActionableList`
- `AdvancedAccountForecast`
- `BenefitManagement`—Available in API version 61.0 and later.
- `BillingSchedulesforInvoiceGeneration`—Billing Schedules for Invoice Generation—Available in API version 62.0 and later.
- `ChannelInventoryManagement`—Available in API version 63.0 and later.
- `CriteriaBsdSearchAndFilter` - Criteria-Based Search And Filter
- `DataProcessingEngine` - Standard
- `DecisionMatrixDataUpload`-This value is available only if you have Business Rules Engine enabled.
- `Decisiontable`—Decision table activation—Available in API version 62.0 and later.
- `EmployeeService`—Available in API version 63.0 and later.
- `FinancialSummaryRollup`—Available in API version 63.0 and later.
- `FSCHierarchyRollUp`
- `FundraisingRollups`—Available in API version 63.0 and later.
- `InvoiceGeneration`—Available in API version 62.0 and later.
- `LegalEntityAccountingPeriodClosureAdvanced`—Available in API version 63.0 and later.
- `LifeSciencesCommercialTerritoryAlignment`—Available in API version 63.0 and later.
- `Loyalty`
- `LoyaltyPartnerManagement`
- `LoyaltyPointsAggregation`
- `NetZero`
- `NextGenForecasting`—Available in API version 62.0 and later.
- `PnmRosterFileUpload`—Available in API version 62.0 and later.
- `PriceProtection`—Available in API version 62.0 and later.
- `ProductCatalogManagement`—Available in API version 63.0 and later.
- `ProgramBasedBusiness`
- `ProgramManagementRollups`
- `Rebates`
- `RevenueTransactionManagement`—Available in API version 63.0 and later.
- `AccountingSubledger`—This value is reserved for internal use.

Field	Details
	<ul style="list-style-type: none"> <li>• <code>ProviderSearch</code>—This value is reserved for internal use.</li> <li>• <code>Recruitment</code>—Available in API version 62.0 and later.</li> <li>• <code>SalesAgreement</code>—Available in API version 63.0 and later.</li> <li>• <code>TestProcessType</code>—Available in API version 62.0 and later.</li> <li>• <code>UsageManagement</code>—Available in API version 62.0 and later.</li> </ul> <p>When Data Processing Engine is enabled for a Salesforce org, the default value is 'Standard'. Other process types may be available to you depending on your industry solution and permission sets.</p>

## Data Processing Engine Metadata API

Use a Metadata API to create, update, and activate Data Processing Engine definitions.

### [BatchCalcJobDefinition](#)

Represents a Data Processing Engine definition.

### BatchCalcJobDefinition

Represents a Data Processing Engine definition.

### Parent Type

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

### File Suffix and Directory Location

`BatchCalcJobDefinition` components have the suffix `.batchCalcJobDefinition` and are stored in the `batchCalcJobDefinitions` folder.

### Version

`BatchCalcJobDefinition` components are available in API version 51.0 and later.

### Special Access Rules

To use this metadata type, one of these licenses is required:

- Loyalty Management
- Financial Services Cloud
- Rebate Management
- Manufacturing Cloud
- Net Zero Cloud

### EDITIONS

Available in: Lightning Experience

Available in: Data Processing Engine is available with Enterprise, Unlimited, and Performance Editions with the Financial Services Cloud, Loyalty Management, Manufacturing Cloud, Rebate Management, Accounting Subledger, or Provider Search in Health Cloud.

## Fields

Field Name	Field Type	Description
aggregates	<a href="#">BatchCalcJobAggregate[]</a>	Collection of aggregate nodes in a data processing engine.
appends	<a href="#">BatchCalcJobUnion[]</a>	Collection of append nodes in a data processing engine.
atomicWritebacks	<a href="#">BatchCalcJobAtomicWriteback[]</a>	Collection of atomic writeback nodes in a data processing engine definition. Available in API version 62.0 and later.
customNodes	<a href="#">BatchCalcJobCustomNode[]</a>	Collection of custom nodes in a data processing engine. Available in API version 57.0 and later.
dataSpaceApiName	string	Stores the Data Space API Name from Data Cloud. Available in API version 60.0 and later.
datasources	<a href="#">BatchCalcJobDatasource[]</a>	Collection of data source nodes in a data processing engine.
definitionRunMode	<a href="#">BatchCalcJobRunModeEnum</a> (of type string)	Specifies the execution mode in a data processing engine. Valid values are: <ul style="list-style-type: none"> <li>• Batch</li> <li>• OnDemand</li> </ul>
description	string	Description of a data processing engine definition.
executionPlatformType	<a href="#">BatchCalcJobExecutionPlatformTypeEnum</a> (of type string)	The platform that's used to run the Data Processing Engine definition. Valid values are: <ul style="list-style-type: none"> <li>• CRMA</li> <li>• CDP</li> <li>• CORE</li> </ul> Available in API version 59.0 and later.
filters	<a href="#">BatchCalcJobFilter[]</a>	Collection of filter nodes in a data processing engine. definition.
forecasts	<a href="#">BatchCalcJobForecast[]</a>	Collection of forecast nodes in a data processing engine. definition. Available in API version 58.0 and later.
hierarchyPaths	<a href="#">BatchCalcJobHierarchyPath[]</a>	Collection of hierarchy path nodes in a data processing engine definition.
isTemplate	boolean	Indicates whether it's a template data processing engine definition.
joins	<a href="#">BatchCalcJobSourceJoin[]</a>	Collection of join nodes in a data processing engine.
label	string	The label of a data processing engine definition.
parameters	<a href="#">BatchCalcJobParameter[]</a>	Collection of input variables in a data processing engine.



Field Name	Field Type	Description
processType	BatchCalcProcessType (enumeration of type string)	<p>The process type of a data processing engine. Valid values are:</p> <ul style="list-style-type: none"> <li>• UsageManagement</li> <li>• AccountingPeriodClosure</li> <li>• AccountingSubledger—This value is reserved for internal use.</li> <li>• ActionableList</li> <li>• AdvancedAccountForecast</li> <li>• BenefitManagement</li> <li>• BillingSchedulesforInvoiceGeneration</li> <li>• CDPEenrichment</li> <li>• ChannelInventoryManagement—Available in API version 63.0 and later.</li> <li>• CriteriaBsdSearchAndFilter</li> <li>• DataProcessingEngine</li> <li>• DecisionMatrixDataUpload</li> <li>• Decisiontable</li> <li>• Education</li> <li>• EmployeeService—Available in API version 63.0 and later.</li> <li>• FinancialSummaryRollup</li> <li>• FSCHierarchyRollUp</li> <li>• FundraisingRollups—Available in API version 63.0 and later.</li> <li>• InvoiceGeneration</li> <li>• Loyalty</li> <li>• LegalEntityAccountingPeriodClosureAdvanced—Available in API version 63.0 and later.</li> <li>• LifeSciencebatchcalesCommercialTerritoryAlignment—Available in API version 63.0 and later.</li> <li>• LoyaltyPartnerManagement</li> <li>• LoyaltyPointsAggregation</li> <li>• NextGenForecasting</li> <li>• NetZero</li> <li>• PnmRosterFileUpload</li> <li>• PriceProtection</li> <li>• ProductCatalogManagement</li> <li>• ProgramBasedBusiness</li> <li>• ProviderSearch—This value is reserved for internal use.</li> <li>• Rebates</li> <li>• Recruitment</li> </ul>

Field Name	Field Type	Description
		<ul style="list-style-type: none"> <li>RevenueTransactionManagement—Available in API version 63.0 and later.</li> <li>SalesAgreement—Available in API version 63.0 and later.</li> <li>TestProcessType</li> </ul>
status	BatchJobDefinition Status (enumeration of type string)	Status of a data processing engine definition. Valid values are: <ul style="list-style-type: none"> <li>Active</li> <li>Inactive</li> </ul>
transforms	BatchCalcJobTransform[]	Collection of data transformation nodes in a data processing engine.
writebacks	BatchCalcJobWriteback Object[]	Collection of writeback objects in which the results of the data processing engine are written back.

## BatchCalcJobAggregate

Represents a collection of fields relating to an aggregate node in a data processing engine.

### Fields

Field Name	Field Type	Description
description	string	Description of an aggregate node.
fields	BatchCalcJob AggregateField[]	Required. Collection of aggregation fields.
groupBy	string[]	Required. Collections of fields used to group data in an aggregate node.
label	string	Required. Label of an aggregate node.
name	string	Required. Name of an aggregate node.
sourceName	string	Required. Name of the source node.

## BatchCalcJobAggregateField

Represents a collection of fields relating to an aggregation field in an aggregate node of a data processing engine.

### Fields

Field Name	Field Type	Description
aggregateFunction	BatchCalcJobAggregateFunction (enumeration of type string)	Required. Function used for aggregation. Valid values are: <ul style="list-style-type: none"> <li>Unique—A count of unique values.</li> </ul>

Field Name	Field Type	Description
		<ul style="list-style-type: none"> <li>• <code>Sum</code>—The sum of all values.</li> <li>• <code>Max</code>—The largest value.</li> <li>• <code>Min</code>—The smallest value.</li> <li>• <code>Avg</code>—The average value, calculated as the mean.</li> <li>• <code>Std</code>—The standard deviation.</li> <li>• <code>Stdp</code>—A standard deviation with population variance.</li> <li>• <code>Var</code>—The variance.</li> <li>• <code>VarP</code>—The variance with population.</li> <li>• <code>Count</code>—The total count of values.</li> </ul>
<code>alias</code>	string	Required. Name that subsequent nodes within the data processing engine use to refer to the aggregate field.
<code>sourceFieldName</code>	string	Required. Source node field on which the aggregate is calculated.

### BatchCalcJobAtomicWriteback

Represents a node in a DPE definition that stores the details about the relationship between the writeback nodes and the atomic writeback operations between the nodes.

Field Name	Field Type	Description
<code>description</code>	string	Description of the atomic writeback object.
<code>label</code>	string	Required. Name of the atomic writeback object.
<code>name</code>	string	Required. API name of the atomic writeback object.
<code>writebackObjectRelationships</code>	<a href="#">BatchAtomicWritebackRelationship</a>	Specifies the relationship between the writeback objects that are involved in the writeback operation.
<code>writebackSequence</code>	int	Sequence in which the data processing engine executes the atomic write back node.

### BatchCalcJobAtomicWritebackRelationship

Represents the relationships between the writeback objects that are involved in an atomic writeback operation. It captures the relationships between these objects and the sequence in which they should be processed.

Field Name	Field Type	Description
<code>childWritebackObjectField</code>	string	Field name that's associated with the child writeback object in an atomic writeback relationship. Available in API version 63.0 and later.
<code>childWritebackObjectName</code>	string	Name of the child writeback object that's associated with the writeback relationship.

Field Name	Field Type	Description
parentWriteback ObjectField	string	Field name that's associated with the parent writeback object in an atomic writeback relationship. Available in API version 63.0 and later.
parentWriteback ObjectName	string	Required. Name of the parent writeback object that's associated with the writeback relationship.
sequenceNumber	int	Sequence number of the writeback node that's associated with its parent node in the relationship.

## BatchCalcJobCustomNode

Represents a collection of custom nodes in a data processing engine. Use a custom node to add a custom action.

### Fields

Field Name	Field Type	Description
description	string	Description of a custom node.
extensionName	string	Required. Name of an extension node.
extensionNamespace	string	Required. Namespace of an extension node.
label	string	Required. Label of a custom node.
name	string	Required. Name of a custom node.
parameters	<a href="#">BatchCalcJobCustomNodeParameter[]</a>	The field mappings of an extension node.
sources	string[]	Sources of an extension node.

## BatchCalcJobCustomNodeParameter

Represents the field mappings of an extension node.

### Fields

Field Name	Field Type	Description
name	string	Required. Name of a parameter.
value	string	Required. Value of a parameter.

## BatchCalcJobDatasource

Represents a collection of fields relating to a data source node in a data processing engine.

## Fields

Field Name	Field Type	Description
<code>CSVDelimiter</code>	<code>BatchCalcJobCSVDelimiter</code> (enumeration of type string)	<p>Specifies the field separator to read fields from a CSV file record.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>• <code>COMMA</code></li> <li>• <code>BACKQUOTE</code></li> <li>• <code>CARET</code></li> <li>• <code>PIPE</code></li> <li>• <code>SEMICOLON</code></li> <li>• <code>TAB</code></li> </ul> <p>The default value is <code>COMMA</code>.</p> <p>The same delimiter value used for the CSV file can't be used within any of the column values in the file. If you mistakenly use the same delimiter value in column values, it can cause data parsing issues.</p>
<code>description</code>	string	Description of a data source node.
<code>fields</code>	<a href="#">BatchCalcJobDataSourceField[]</a>	Required. Collection of data source fields.
<code>fileIdentifier</code>	string	Specifies the source of the file or file storage system.
<code>filePath</code>	string	The file path for the specified file.
<code>fileSource</code>	<code>BatchCalcJobFileSource</code> (enumeration of type string)	<p>Specifies the source of the file or file storage system.</p> <p>Possible value is:</p> <ul style="list-style-type: none"> <li>• <code>ContentManagement</code></li> </ul>
<code>label</code>	string	Required. Label of a data source node.
<code>name</code>	string	Required. Name of a data source node.
<code>sourceName</code>	string	Required. Name of a standard or custom object from which the data source node extracts data.
<code>type</code>	<code>BatchCalcJobDataSourceType</code> (enumeration of type string)	<p>Required. Type of object for the source object field. Supported values are:</p> <ul style="list-style-type: none"> <li>• <code>Analytics</code></li> <li>• <code>CalculatedInsightsObject</code></li> <li>• <code>CRMObject</code></li> <li>• <code>CSV</code></li> <li>• <code>DataModelObject</code></li> <li>• <code>StandardObject</code></li> </ul>

## BatchCalcJobDatasourceField

Represents a collection of fields relating to a source object field that are selected in the data source node of a data processing engine.

### Fields

Field Name	Field Type	Description
alias	string	Name that subsequent nodes within the data processing engine use to refer to the data source field. Required when the field name is lookup.
dataType	BatchCalcJobDataType (enumeration of type string)	Specifies the data type of the input field when using a CSV file as a data source. Possible values are: <ul style="list-style-type: none"> <li>• Text</li> <li>• Numeric</li> <li>• Date</li> <li>• DateTime</li> <li>• MultiValue</li> </ul>
isPrimaryKey	boolean	Indicates whether a column name is the primary key ( <code>true</code> ) or not ( <code>false</code> ) for the Data Cloud CSV file.
name	string	Required. Name of the field. Can be either of the following: <ul style="list-style-type: none"> <li>• Name of the source field selected in the associated data source object.</li> <li>• Name from a nested lookup object with three child levels.</li> </ul>

## BatchCalcJobFilter

Represents a collection of fields relating to a filter node in a data processing engine.

### Fields

Field Name	Field Type	Description
criteria	BatchCalcJobFilter Criteria[]	Collection of filter criteria in a filter node. The field is required when <code>isDynamicFilter</code> is set to <code>False</code> .
description	string	Description of the batch calculation job filter.
filterCondition	string	Logic that is specified to apply the filter conditions. The field is required when <code>isDynamicFilter</code> is set to <code>False</code> .
filterParameterName	string	Name of the parameter of type filter.
isDynamicFilter	boolean	Indicates whether the filter criteria is dynamic. If value is set to <code>True</code> , filter criteria is passed in runtime with <code>filterParameterName</code> .
label	string	Required. Label of the filter node.

Field Name	Field Type	Description
name	string	Required. Name of the filter node.
sourceName	string	Required. Name of the source node.

## BatchCalcJobForecast

Represents a collection of fields relating to a forecast node in a data processing engine. Available in API version 58.0 and later.

### Fields

Field Name	Field Type	Description
<del>accuracy</del> <code>BatchCalcJobFrcstAccuracy</code> (enumeration of type string)		The interval percentage to account for errors in forecasts.  Possible values are: <ul style="list-style-type: none"> <li>Eighty</li> <li>NinetyFive</li> <li>None</li> </ul> The default value is <code>None</code> .
<del>aggrFields</del> <code>BatchCalcJobFrcstAggrFld[]</code>		The list of fields to forecast.
<del>dateField</del> string		Required.  The date field from the source node used to forecast values for the specified forecast length.
<del>description</del> string		The description of the forecast node.
<del>forecastModel</del> <code>BatchCalcJobFrcstModel</code> (enumeration of type string)		The model used to forecast data.  Possible values are: <ul style="list-style-type: none"> <li>Additive</li> <li>Auto</li> <li>Multiplicative</li> </ul> The default value is <code>Auto</code> .
<del>forecastPeriod</del> int		The number of time periods to generate

Field Name	Field Type	Description
<del>forecastPeriod</del>	BatchCalcJobFrcstPeriodType (enumeration of type string)	<p>forecast data. For example, if you select Year-Month as the forecast period type, and 4 as the forecast period count, the forecast results are generated for the next 4 months.</p> <p>The minimum and the default count is 1, and the maximum is 100.</p>
<del>forecastPeriod</del>	BatchCalcJobFrcstPeriodType (enumeration of type string)	<p>Required.</p> <p>The type of forecast period to group date field values in the forecast results.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>• FiscalYear</li> <li>• FiscalYearMonth</li> <li>• FiscalYearQuarter</li> <li>• FiscalYearWeek</li> <li>• Year</li> <li>• YearMonth</li> <li>• YearMonthDay</li> <li>• YearQuarter</li> <li>• YearWeek</li> </ul>
<del>groupFields</del>	BatchCalcJobFrcstGrpFld[]	The source fields for grouping the data to be processed by the forecast node.
label	string	<p>Required.</p> <p>The name of the forecast node in the UI.</p>
name	string	<p>Required.</p> <p>A unique name for the forecast node.</p>
<del>startDate</del>	string	<p>Required.</p> <p>The start date of the forecast period.</p>



Field Name	Field Type	Description
<del>seasonality</del>	BatchCalcJobFrstSeasonality (enumeration of type string)	<p>Represents the periodic fluctuations that occur around the same time every year.</p> <p>Possible values are:</p> <ul style="list-style-type: none"><li>• Two</li><li>• Three</li><li>• Four</li><li>• Five</li><li>• Six</li><li>• Seven</li><li>• Eight</li><li>• Nine</li><li>• Ten</li><li>• Eleven</li><li>• Twelve</li><li>• Thirteen</li><li>• Fourteen</li><li>• Fifteen</li><li>• Sixteen</li><li>• Seventeen</li><li>• Eighteen</li><li>• Nineteen</li><li>• Twenty</li><li>• TwentyOne</li><li>• TwentyTwo</li><li>• TwentyThree</li><li>• TwentyFour</li><li>• Auto</li><li>• None</li></ul> <p>The default value is None.</p>
<del>ignoreLastPeriod</del>	boolean	<p>Indicates whether to ignore the last period in the source node when it has incomplete data (<code>true</code>) or not (<code>false</code>).</p> <p>The default value is <code>false</code>.</p>

Field Name	Field Type	Description
sourceName	string	Required. The name of the source node. A source can be any node other than the datasink and register node.

### BatchCalcJobFrcstAggrFld

Represents a list of fields to forecast in a forecast node.

Field Name	Field Type	Description
aggregateFunction	BatchCalcJobAggregateFunction (enumeration of type string)	Required. The function of the aggregate field. Possible values are: <ul style="list-style-type: none"> <li>• Avg</li> <li>• Count</li> <li>• Max</li> <li>• Min</li> <li>• Std</li> <li>• StdP</li> <li>• Sum</li> <li>• Unique</li> <li>• Var</li> <li>• VarP</li> </ul>
aggregationResultLabel	string	Required. The name of the aggregation result generated from the aggregation function that's applied to the source node field.
fieldName	string	Required. The name of the source field.

### BatchCalcJobFrcstGrpFld

Represents source fields for grouping the data to be processed by the forecast node.

Field Name	Field Type	Description
fieldName	string	Required. The name of the source field to group the data to be processed by the forecast node.
groupBy	string	A comma-separated list of values to group data by. Required when the source field type is Date or DateTime. Possible values are: <ul style="list-style-type: none"><li>• Second</li><li>• Second Epoch</li><li>• Minute</li><li>• Hour</li><li>• Day</li><li>• Day Epoch</li><li>• Week</li><li>• </li><li>• Month</li><li>• Quarter</li><li>• Year</li></ul>

## BatchCalcJobHierarchyPath

Represents a collection of hierarchy path nodes in a data processing engine definition.

### Fields

Field Name	Field Type	Description
description	string	Description of the hierarchy path node.
hierarchyFieldName	string	Required. Field name that contains the hierarchy path.
isSelfFieldValueIncluded	boolean	Indicates whether the self value is included in the calculated hierarchy path ( <code>True</code> ) or not ( <code>False</code> ).
label	string	Required. Label of the hierarchy path node.
name	string	Required. Name of the hierarchy path node.
parentFieldName	string	Required. Parent field name to calculate hierarchy path.
selfFieldName	string	Required. Self field name to calculate hierarchy path.
sourceName	string	Required. Name of the source node.

## BatchCalcJobFilterCriteria

Represents a collection of fields relating to a filter condition in a filter node in a data processing engine.

### Fields

Field Name	Field Type	Description
<code>inputVariable</code>	string	Name of the input variable used as a filter.
<code>operator</code>	BatchCalcJobFilter Operator (enumeration of type string)	Required. Operator that is specified in the filter condition.  Valid values are: <ul style="list-style-type: none"><li>• Equals</li><li>• NotEquals</li><li>• GreaterThan</li><li>• GreaterThanOrEqual</li><li>• LessThan</li><li>• LessThanOrEqual</li><li>• StartsWith</li><li>• EndsWith</li><li>• Contains</li><li>• DoesNotContain</li><li>• IsNull</li><li>• IsNotNull</li><li>• In</li><li>• NotIn</li></ul>
<code>sequence</code>	integer	Required. Sequence number used to refer the criteria in a filter node.
<code>sourceFieldName</code>	string	Required. Name of the field from the source node to apply the filter.
<code>value</code>	string	Value used to filter data from the source node.

## BatchCalcJobParameter

Represents a collection of fields relating to an input variable in a data processing engine.

### Fields

Field Name	Field Type	Description
<code>dataType</code>	BatchCalcJobParameter DataType (enumeration of type string)	Required. Data type of the parameter. Valid values are: <ul style="list-style-type: none"><li>• Date</li><li>• DateTime</li><li>• Expression</li></ul>

Field Name	Field Type	Description
		<ul style="list-style-type: none"> <li>• <code>FileIdentifier</code></li> <li>• <code>Filter</code></li> <li>• <code>Numeric</code></li> <li>• <code>Text</code></li> </ul>
<code>defaultValue</code>	string	Default value of the parameter.
<code>description</code>	string	Description of the batch calculation job parameter.
<code>isMultiValue</code>	boolean	Indicates whether the parameter has different values ( <code>True</code> ) or not ( <code>False</code> ). This field is supported only for the <code>Text</code> data type.
<code>label</code>	string	Required. Label of the batch calculation job parameter.
<code>name</code>	string	Required. Name of the batch calculation job parameter.

## BatchCalcJobSourceJoin

Represents a collection of fields relating to a join node in a data processing engine.

### Fields

Field Name	Field Type	Description
<code>description</code>	string	Description of the join node.
<code>fields</code>	<a href="#">BatchCalcJobJoinResultField[]</a>	Collection of fields in a join node.
<code>joinKeys</code>	<a href="#">BatchCalcJobJoinKey[]</a>	Collection of mapping of fields from the primary source node and the second source node in a join node.
<code>label</code>	string	Required. Label of the join node.
<code>name</code>	string	Required. Name of the join node.
<code>primarySourceName</code>	string	Required. Name associated with the node as the primary source node.
<code>secondarySourceName</code>	string	Required. Name associated with the node as the secondary source node.
<code>type</code>	<a href="#">BatchCalcJobSourceJoinType</a> (enumeration of type string)	Required. Type of join specified between the primary source node and secondary source node. Valid values are: <ul style="list-style-type: none"> <li>• <code>LeftOuter</code></li> <li>• <code>RightOuter</code></li> <li>• <code>Inner</code></li> <li>• <code>Outer</code></li> <li>• <code>Lookup</code></li> </ul>

## BatchCalcJobJoinKey

Represents a collection of fields relating to a mapping of fields from the first source node and second source node in a join node of a data processing engine.

### Fields

Field Name	Field Type	Description
primarySourceFieldName	string	Required. Mapped field name of the primary source node.
secondarySourceFieldName	string	Required. Mapped field name of the secondary source node.

## BatchCalcJobJoinResultField

Represents a collection of fields relating to a set of resultant fields in a join node of a data processing engine.

### Fields

Field Name	Field Type	Description
alias	string	Required. Name that subsequent nodes within the data processing engine definition use to refer to the resultant field.
sourceFieldName	string	Required. Name of field from the primary or secondary data source.
sourceName	string	Required. Source node of the primary or secondary data source.

## BatchCalcJobTransform

Represents a collection of fields relating to a data transformation in a data processing engine.

### Fields

Field Name	Field Type	Description
description	string	The description of the batch calculation job transform.
droppedFields	<a href="#">BatchCalcJobTransformDroppedField[]</a>	The collection of dropped fields in a data transformation. Available when the transformation type is <code>SLICE</code> .
expressionFields	<a href="#">BatchCalcJobTransformAddedField[]</a>	The collection of formula fields in a data transformation. Available when the transformation type is <code>EXPRESSION</code> .
label	string	Required. The label of the batch calculation job transform.
name	string	Required. The name of the batch calculation job transform.
orderBy	<a href="#">BatchCalcJobOrderByField</a> on page 680[]	A collection of fields that's used to sort the records within each partition group.
partitionBy	string[]	A group of fields that's used to partition the source data into partition groups.

Field Name	Field Type	Description
sourceName	string	Required. Name of the source node.
transformType	BatchCalcJobTransformType (enumeration of type string)	Required. The type of transformation. Valid values are: <ul style="list-style-type: none"> <li>• <code>ComputeRelative</code>—This transformation calculates values based on values of the same partition group.</li> <li>• <code>Expression</code>—This transformation calculates values based on existing values of fields in the same record.</li> <li>• <code>Slice</code>—This transformation removes fields from the source node.</li> </ul>

### BatchCalcJobTransformDroppedField

Represents a collection of fields relating to a dropped field in a data transformation of a data processing engine.

#### Fields

Field Name	Field Type	Description
sourceFieldName	string	Required. Name of the field that is dropped.

### BatchCalcJobTransformAddedField

Represents a collection of fields relating to a formula in a data transformation of a data processing engine.

#### Fields

Field Name	Field Type	Description
alias	string	Required. Name that subsequent nodes within the data processing engine use to the transform node.
dataType	BatchCalcJobDataType (enumeration of type string)	Required. Data type of the formula. Valid values are: <ul style="list-style-type: none"> <li>• <code>Text</code></li> <li>• <code>Numeric</code></li> <li>• <code>Date</code></li> <li>• <code>DateTime</code></li> <li>• <code>MultiValue</code></li> </ul>
decimalPlaces	integer	Number of digits to the right of a decimal point in the value. Required for the <code>Numeric</code> data type.
expression	string	Required. Formula defined by the user.

Field Name	Field Type	Description
length	integer	Total length of the value including the decimal places. Required for data types: <code>Text</code> and <code>Numeric</code> .

## BatchCalcJobOrderByField

Represents a collection of fields that are used to sort the partitioned data.

### Fields

Field Name	Field Type	Description
name	string	Required. Name of the field that is used to sort data.
orderType	BatchCalcJobOrderType (enum of type string)	Order in which the data is sorted. Valid values are: <ul style="list-style-type: none"> <li>Ascending</li> <li>Descending</li> </ul>

## BatchCalcJobUnion

Represents a collection of fields relating to the union of data from two nodes in a data processing engine.

### Fields

Field Name	Field Type	Description
description	string	Description of the batch calculation job union.
isDisjointedSchema	boolean	Indicates whether the union is of two disjointed datasets ( <code>true</code> ) or not ( <code>false</code> ). Set to <code>True</code> to allow joining of two datasets having no common fields.
label	string	Required. Label of the batch calculation job union.
name	string	Required. Name of the batch calculation job union.
sources	string[]	Names of the source nodes.

## BatchCalcJobWritebackObject

Represents a collection of fields relating to the object in which the results of the data processing engine are written back.



## Fields

Field Name	Field Type	Description
description	string	Descriptions of the batch calculation job writeback object.
externalIdFieldName	string	Unique external field ID for the target object name. Available in API version 60.0 and later.
fields	<a href="#">BatchCalcJobWriteback Mapping[]</a>	Collection of the writeback fields.
filterCondition	string	The condition that filters the records from a writeback dataset for a user. Examples of a filter condition include a user ID, stage name, and a security policy that returns only the records that a user owns. Available in API version 57.0 and later.
folderName	string	The folder where the writeback dataset is saved. Available in API version 57.0 and later.
isChangedRow	boolean	Indicates whether a row in the write back object is changed. Set to <code>True</code> to write back the changed rows.
isExistingDataset	boolean	Indicates whether a CRM Application (CRMA) dataset or a Data Cloud Data Lake object is present ( <code>true</code> ) or will be created ( <code>false</code> ). Available in API version 62.0 and later.
label	string	Required. Name of the write back object.
name	string	Required. Name of the batch calculation job write back object.
operationType	BatchCalcJobWriteback Opn (enumeration of type string)	Type of operation specified. Valid values are: <ul style="list-style-type: none"> <li>• <code>Delete</code>—This value is available in API version 56.0 and later.</li> <li>• <code>Insert</code></li> <li>• <code>Overwrite</code>—Available only when <code>storageType</code> is <code>DataLakeObject</code>. This value is available in API version 60.0 and later.</li> <li>• <code>Update</code></li> <li>• <code>Upsert</code></li> </ul>
sharingInheritanceObjectName	string	The name of the source object from which the row-level sharing inheritance settings are applied. Available in API version 57.0 and later.
sourceName	string	Required. Name of the source node associated with the write back object.
storageType	BatchCalcJobWriteback Type (enumeration of type string)	Specifies where you want to use the data stored in the source node. Available in API version 57.0 and later. Valid values are: <ul style="list-style-type: none"> <li>• <code>Analytics</code></li> <li>• <code>DataLakeObject</code></li> </ul>

Field Name	Field Type	Description
		<ul style="list-style-type: none"> <li>sObject</li> </ul> The default value is sObject.
targetObjectName	string	Required. Object that is inserted or upserted by the data processing engine.
writebackSequence	integer	Sequence in which the target object is updated by the data processing engine.
writebackUser	string	ID of the user whose permissions decide which objects and fields of the target object can be updated.

## BatchCalcJobWritebackMapping

Represents a collection of fields relating to the mapping between results and the fields in the target object.

### Fields

Field Name	Field Type	Description
parentName	string	Name of the lookup object. Required only when the relationshipName field is defined.
relationshipName	string	Name of the lookup relationship.
runtimeParameter	boolean	Indicates whether the source field from runtime parameter is true or false. The default value is false. Available in API version 59.0 and later.
sourceFieldName	string	Required. Name of the field in the source node that is written back.
targetFieldName	string	Name of the sObject field to which the results are written back.

## Declarative Metadata Sample Definition

The following is an example of a BatchCalcJobDefinition component.

```
<?xml version="1.0" encoding="UTF-8"?>
<BatchCalcJobDefinition xmlns="http://soap.sforce.com/2006/04/metadata">
  <aggregates>
    <description>Aggregate Description</description>
    <fields>
      <aggregateFunction>Count</aggregateFunction>
      <alias>NameCount</alias>
      <sourceFieldName>Name</sourceFieldName>
    </fields>
    <groupBy>ContactId</groupBy>
    <groupBy>Name</groupBy>
    <label>AggregateOpportunities</label>
    <name>AggregateOpportunities</name>
    <sourceName>Opportunity</sourceName>
  </aggregates>
</BatchCalcJobDefinition>
```

```
</aggregates>
<forecasts>
  <description>ForecastNode Description</description>
  <label>ContactForecast</label>
  <name>ContactForecast</name>
  <sourceName>Contact</sourceName>
  <dateFieldName>CreateDate</dateFieldName>
  <forecastPeriodType>YearMonth</forecastPeriodType>
  <shouldExcludeLastPeriod>false</shouldExcludeLastPeriod>
  <forecastPeriodCount>12</forecastPeriodCount>
  <periodStartDateName>CreateDateYM</periodStartDateName>
  <forecastModelType>Auto</forecastModelType>
  <seasonality>None</seasonality>
  <accuracyPercent>None</accuracyPercent>
  <aggregationFields>
    <aggregateFunction>Count</aggregateFunction>
    <aggregationResultLabel>CountOfLastName</aggregationResultLabel>
    <fieldName>LastName</fieldName>
  </aggregationFields>
  <groupFields>
    <fieldName>LastModifiedDate</fieldName>
    <groupBy>Week</groupBy>
  </groupFields>
</forecasts>
<appends>
  <description>Append desc</description>
  <isDisjointedSchema>true</isDisjointedSchema>
  <label>AppendAllAccounts</label>
  <name>AppendAllAccounts</name>
  <sources>AccountsOfManufacturingIndustry</sources>
  <sources>ComputeRelativeManufacturingIndustry</sources>
</appends>
<datasources>
  <description>Desc Contact</description>
  <fields>
    <alias>Id</alias>
    <name>Id</name>
    <isPrimaryKey>false</isPrimaryKey>
    <dataType>Text</dataType>
  </fields>
  <fields>
    <alias>LastName</alias>
    <name>LastName</name>
    <isPrimaryKey>false</isPrimaryKey>
    <dataType>Text</dataType>
  </fields>
  <fields>
    <alias>CreateDate</alias>
    <name>CreateDate</name>
    <isPrimaryKey>false</isPrimaryKey>
    <dataType>Date</dataType>
  </fields>
  <fields>
    <alias>LastModifiedDate</alias>
```

```
    <name>LastModifiedDate</name>
    <isPrimaryKey>>false</isPrimaryKey>
    <dataType>Date</dataType>
</fields>
<label>Contact</label>
<name>Contact</name>
<sourceName>Contact</sourceName>
<type>StandardObject</type>
<fileSource>ContentManagement</fileSource>
<fileIdentifier>069xx0000004CAeAAM</fileIdentifier>
<CSVDelimitter>COMMA</CSVDelimitter>
<filePath>parentFolder/childFolder</filePath>
</datasources>
<datasources>
  <fields>
    <alias>Name</alias>
    <name>Name</name>
    <isPrimaryKey>>false</isPrimaryKey>
    <dataType>Text</dataType>
  </fields>
  <fields>
    <alias>ContactId</alias>
    <name>ContactId</name>
    <isPrimaryKey>>false</isPrimaryKey>
    <dataType>Text</dataType>
  </fields>
  <label>Opportunity</label>
  <name>Opportunity</name>
  <sourceName>Opportunity</sourceName>
  <type>StandardObject</type>
  <fileSource>ContentManagement</fileSource>
  <fileIdentifier>069xx0000004CAeAAM</fileIdentifier>
  <CSVDelimitter>COMMA</CSVDelimitter>
  <filePath>parentFolder/childFolder</filePath>
</datasources>
<description>Calculates and creates transaction journal records based on the orders
placed by the loyalty program members. The transaction journals are used to accrue points
to the member.</description>
<filters>
  <criteria>
    <operator>Equals</operator>
    <sequence>1</sequence>
    <sourceFieldName>LastName</sourceFieldName>
    <value>Salesforce</value>
  </criteria>
  <description>Filter Desc</description>
  <filterCondition>1</filterCondition>
  <isDynamicFilter>>false</isDynamicFilter>
  <label>AccountsOfManufacturingIndustry</label>
  <name>AccountsOfManufacturingIndustry</name>
  <sourceName>AccountOpportunities</sourceName>
</filters>
<hierarchyPaths>
  <description>Hierarchy Path Node</description>
```

```

    <hierarchyFieldName>Hierarchy_Path</hierarchyFieldName>
    <isSelfFieldValueIncluded>true</isSelfFieldValueIncluded>
    <label>Get Hierarchy</label>
    <name>Get_Hierarchy</name>
    <parentFieldName>ContactId</parentFieldName>
    <selfFieldName>LastName</selfFieldName>
    <sourceName>AppendAllAccounts</sourceName>
  </hierarchyPaths>
  <isTemplate>>false</isTemplate>
  <joins>
    <description>Left Outer Join</description>
    <fields>
      <alias>ContactId</alias>
      <sourceFieldName>Id</sourceFieldName>
      <sourceName>Contact</sourceName>
    </fields>
    <fields>
      <alias>LastName</alias>
      <sourceFieldName>LastName</sourceFieldName>
      <sourceName>Contact</sourceName>
    </fields>
    <fields>
      <alias>NameCount</alias>
      <sourceFieldName>NameCount</sourceFieldName>
      <sourceName>AggregateOpportunities</sourceName>
    </fields>
    <fields>
      <alias>OpportunityName</alias>
      <sourceFieldName>Name</sourceFieldName>
      <sourceName>AggregateOpportunities</sourceName>
    </fields>
    <joinKeys>
      <primarySourceFieldName>Id</primarySourceFieldName>
      <secondarySourceFieldName>ContactId</secondarySourceFieldName>
    </joinKeys>
    <label>AccountOpportunities</label>
    <name>AccountOpportunities</name>
    <primarySourceName>Contact</primarySourceName>
    <secondarySourceName>AggregateOpportunities</secondarySourceName>
    <type>LeftOuter</type>
  </joins>
  <label>Create Transaction Journals Based on Orders</label>
  <parameters>
    <dataType>Date</dataType>
    <defaultValue>2020-01-01</defaultValue>
    <description>Desc TextParameter</description>
    <isMultiValue>>false</isMultiValue>
    <label>DateParameter</label>
    <name>DateParameter</name>
  </parameters>
  <parameters>
    <dataType>Filter</dataType>
    <defaultValue>{&quot;filterCondition&quot;: &quot;1 AND 2&quot;,
&quot;criteria&quot;: [{&quot;sourceFieldName&quot;:

```

```
&quot;NameCount&quot;;,&quot;operator&quot;: &quot;GreaterThan&quot;;,&quot;value&quot;:
&quot;20&quot;;,&quot;sequence&quot;: &quot;1&quot;}}, {&quot;sourceFieldName&quot;:
&quot;Name&quot;;,&quot;operator&quot;: &quot;Equals&quot;;,&quot;value&quot;:
&quot;Salesforce&quot;;,&quot;sequence&quot;: &quot;2&quot;}}</defaultValue>
  <isMultiValue>>false</isMultiValue>
  <label>FilterParameter</label>
  <name>FilterParameter</name>
</parameters>
<parameters>
  <dataType>Numeric</dataType>
  <defaultValue>5000</defaultValue>
  <description>Desc TextParameter</description>
  <isMultiValue>>false</isMultiValue>
  <label>NumericParameter</label>
  <name>NumericParameter</name>
</parameters>
<parameters>
  <dataType>Text</dataType>
  <defaultValue>@salesforce.com</defaultValue>
  <description>Desc TextParameter</description>
  <isMultiValue>>false</isMultiValue>
  <label>TextParameter</label>
  <name>TextParameter</name>
</parameters>
<processType>Rebates</processType>
<definitionRunMode>Batch</definitionRunMode>
<status>Inactive</status>
<transforms>
  <description>transforms Desc</description>
  <expressionFields>
    <alias>NewLastName</alias>
    <dataType>Text</dataType>
    <expression>TODAY ()</expression>
    <length>80</length>
  </expressionFields>
  <label>ManufacturingIndustry</label>
  <name>ManufacturingIndustry</name>
  <sourceName>AccountsOfManufacturingIndustry</sourceName>
  <transformationType>Expression</transformationType>
</transforms>
<transforms>
  <droppedFields>
    <sourceFieldName>NewLastName</sourceFieldName>
  </droppedFields>
  <label>MediaIndustry</label>
  <name>MediaIndustry</name>
  <sourceName>ManufacturingIndustry</sourceName>
  <transformationType>Slice</transformationType>
</transforms>
<transforms>
  <description>compute relative transforms Desc</description>
  <expressionFields>
    <alias>NewLastName</alias>
    <dataType>Text</dataType>
```

```

        <expression>rank ()</expression>
        <length>80</length>
    </expressionFields>
    <label>ComputeRelativeManufacturingIndustry</label>
    <name>ComputeRelativeManufacturingIndustry</name>
    <orderBy>
        <name>LastName</name>
        <orderType>Ascending</orderType>
    </orderBy>
    <partitionBy>LastName</partitionBy>
    <sourceName>MediaIndustry</sourceName>
    <transformationType>ComputeRelative</transformationType>
</transforms>
<customNodes>
    <name>RebatesCustomNode</name>
    <label>Rebates Custom Node</label>
    <description>customNodes Desc</description>
    <sources>Get_Hierarchy</sources>
    <extensionName>RebatesExpression</extensionName>
    <extensionNamespace>industries_mfg</extensionNamespace>
    <parameters>
        <name>inputColumn</name>
        <value>LastName</value>
    </parameters>
    <parameters>
        <name>isFilterCriteria</name>
        <value>>true</value>
    </parameters>
    <parameters>
        <name>outputColumn</name>
        <value>GenName</value>
    </parameters>
</customNodes>
<writebacks>
    <fields>
        <sourceFieldName>GenName</sourceFieldName>
        <targetFieldName>LastName</targetFieldName>
    </fields>
    <isChangedRow>>false</isChangedRow>
    <label>exportToContact</label>
    <name>exportToContact</name>
    <description>Export To Contact</description>
    <operationType>Insert</operationType>
    <sourceName>RebatesCustomNode</sourceName>
    <targetObjectName>Contact</targetObjectName>
    <writebackSequence>1</writebackSequence>
</writebacks>
<writebacks>
    <fields>
        <sourceFieldName>CreateDateYM</sourceFieldName>
        <targetFieldName>CreateDate</targetFieldName>
    </fields>
    <isChangedRow>>false</isChangedRow>
    <isExistingDataset>>false</isExistingDataset>

```

```

    <label>exportToContactFC</label>
    <name>exportToContactFC</name>
    <description>Export To Contact</description>
    <operationType>Insert</operationType>
    <sourceName>ContactForecast</sourceName>
    <targetObjectName>Contact</targetObjectName>
    <writebackSequence>2</writebackSequence>
  </writebacks>
</BatchCalcJobDefinition>

```

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

```

<?xml version="1.0" encoding="UTF-8"?>
<!--
  ~ Copyright 2020 Salesforce, Inc.
  ~ All Rights Reserved
  ~ Company Confidential
-->
<Package xmlns="http://soap.sforce.com/2006/04/metadata">
  <types>
    <members>*</members>
    <name>BatchCalcJobDefinition</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](#).

## Data Processing Engine Invocable Actions

Run an active Data Processing Engine definition. For more information on custom invocable actions, see [REST API Developer Guide](#) and [Actions Developer Guide](#).

### Data Processing Engine Actions

Run an active Data Processing Engine definition. This action executes a Data Processing Engine definition asynchronously.

## Data Processing Engine Actions

Run an active Data Processing Engine definition. This action executes a Data Processing Engine definition asynchronously.

A Data Processing Engine definition transforms data from your Salesforce org and writes back the results to your org. For more information about running Data Processing Engine definitions, see [Run a Data Processing Engine Definition](#) in Salesforce Help.

This object is available in API version 51.0 and later.

## Supported REST HTTP Methods

### URI

`/services/data/vXX.X/actions/custom/dataProcessingEngineAction`



**Formats**

JSON

**HTTP Methods**

GET, POST

**Authentication**Authorization: Bearer *token***Inputs**

Use the GET method to retrieve input variables of a Data Processing Engine definition. The input variables of each Data Processing Engine definition are unique. The Data Processing Engine action uses the input variables to execute the Data Processing Engine definition and generate a batch job ID.



**Note:** You can't use this action to start two simultaneous runs of a definition with the same input variables.

**Outputs**

Output	Details
batchJobId	<b>Type</b> string  <b>Description</b> ID of the batch job generated after the request is successful. This ID is used to track the progress of the batch job in Monitor Workflow Services.

**Usage****JSON Sample Request to execute PointsAccrual DPE Definition**

```
{
  "PointsAccrual" : {
    "memberTier" : "Gold",
    "minimumPointBalanceRequired" : "50000",
    "pointType" : "non-qualifying"
  }
}
```

**JSON Sample Response**

```
{
  "actionName":"PointsAccrual",
  "errors":null,
  "isSuccess":true,
  "outputValues":{
    "batchJobId":"01Mxx0000A000001EAA"
  }
}
```

## Example

### GET

This example shows how to retrieve input variables of a Data Processing Engine action type.

```
curl --include --request GET \  
--header "Authorization: Authorization: Bearer 00DR...xyz" \  
--header "Content-Type: application/json" \  
"https://instance.salesforce.com/services/data/v60.0/actions/custom/dataProcessingEngineAction/newinputvardefn"
```

### POST

Here's a request to retrieve DPE definition

```
{  
  "inputs": [  
    {  
      "start_date": "26-09-2023",  
      "end_date": "12-12-2023",  
      "randomkey": "069SM0000001SgbYAE"  
    }  
  ]  
}
```

Here's a response for this action.

```
[  
  {  
    "actionName": "newinputvardefn",  
    "errors": null,  
    "invocationID": null,  
    "isSuccess": true,  
    "outputValues": {  
      "batchJobId": "0mdSM0000006EJdYAM",  
      "accepted": true  
    },  
    "version": 1  
  }  
]
```

# Batch Management

Automate the processing of records in scheduled flows. You can process a high volume of standard and custom object records. Batch Management consists of three Tooling API objects, a standard object, a Metadata API, and an invocable action. You can use these resources to view, create, edit, and run Batch Management jobs.

## [Batch Management Tooling API Objects](#)

You can use the Batch Management Tooling API object to view the settings of Batch Management jobs.

## [Batch Management Standard Objects](#)

Batch Management contains a standard BatchProcessJobDefinitionView object, which you can use to view all the batch jobs available in your Salesforce org, including file-based jobs.

## [Batch Management Metadata API](#)

Use a Metadata API to create, update, and activate Batch Management jobs.

## [Batch Management Invocable Actions](#)

Run an active Batch Management job definition. For more information on custom invocable actions, see [REST API Developer Guide](#) and [Actions Developer Guide](#).

### EDITIONS

Available in: Lightning Experience

Available in: Batch Management is available with Enterprise, Unlimited, and Performance Editions with the Loyalty Management, Manufacturing Cloud, Rebate Management, or Accounting Subledger

# Batch Management Tooling API Objects

You can use the Batch Management Tooling API object to view the settings of Batch Management jobs.


Tooling API objects let you interact with metadata for declarative development. For example, you can create your own version of Setup.

## [BatchProcessJobDefinition](#)

Represents the details of a Batch Management job. This object is available in API version 51.0 and later.

# BatchProcessJobDefinition

Represents the details of a Batch Management job. This object is available in API version 51.0 and later.

 **Note:** 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.

Data and processes in your org are impacted if you update or delete a BatchProcessJobDefinition record. Update or delete a Batch Management job using the [Metadata API](#) on page 699.

# Supported SOAP API Calls

`describeSObjects()`, `query()`, `retrieve()`

### EDITIONS

Available in: Lightning Experience

Available in: Batch Management is available with Enterprise, Unlimited, and Performance Editions with the Loyalty Management, Manufacturing Cloud, Rebate Management, or Accounting Subledger

## Supported REST API Methods

DELETE, GET, HEAD, PATCH, POST, Query

## Fields

Field	Details
BatchJobDefinitionId	<p><b>Type</b> reference</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The unique identifier of the associated batch job definition. This is a relationship field.</p> <p><b>Relationship Name</b> BatchJobDefinition</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> BatchJobDefinition</p>
BatchJobDefinitionName	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The name of the associated batch job definition.</p>
BatchSize	<p><b>Type</b> int</p> <p><b>Properties</b> Filter, Group, Sort</p> <p><b>Description</b> Required. The number of records that each Batch Management job part can process. The maximum number of transaction journal records that a batch management job can process for flow or loyalty program process is 2000.</p>
Description	<p><b>Type</b> textarea</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The description of the Batch Management job.</p>

Field	Details
DeveloperName	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Sort</p> <p><b>Description</b> The developer name of the Batch Management job.</p>
FlowDefinitionId	<p><b>Type</b> reference</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The input variable of the associated flow that uniquely identifies each record that the Batch Management job processes.</p>
FullName	<p><b>Type</b> string</p> <p><b>Properties</b> Create, Group, Nillable</p> <p><b>Description</b> The name of the Batch Management job.  Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
Language	<p><b>Type</b> picklist</p> <p><b>Properties</b> Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p><b>Description</b> The language in which the batch job is created.  Possible values are:</p> <ul style="list-style-type: none"> <li>• da—Danish</li> <li>• de—German</li> <li>• en_US—English</li> <li>• es—Spanish</li> <li>• es_MX—Spanish (Mexico)</li> <li>• fi—Finnish</li> <li>• fr—French</li> <li>• it—Italian</li> </ul>

Field	Details
	<ul style="list-style-type: none"> <li>• ja—Japanese</li> <li>• ko—Korean</li> <li>• nl_NL—Dutch</li> <li>• no—Norwegian</li> <li>• pt_BR—Portuguese (Brazil)</li> <li>• ru—Russian</li> <li>• sv—Swedish</li> <li>• th—Thai</li> <li>• zh_CN—Chinese (Simplified)</li> <li>• zh_TW—Chinese (Traditional)</li> </ul>
<p>ManageableState</p>	<p><b>Type</b> ManageableState enumerated list</p> <p><b>Properties</b> Filter, Group, Nillable, Restricted picklist, Sort</p> <p><b>Description</b> Indicates the manageable state of the specified component that is contained in a package:</p> <ul style="list-style-type: none"> <li>• beta</li> <li>• deleted</li> <li>• deprecated</li> <li>• deprecatedEditable</li> <li>• installed</li> <li>• installedEditable</li> <li>• released</li> <li>• unmanaged</li> </ul>
<p>MasterLabel</p>	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Sort</p> <p><b>Description</b> The label of the Batch Management job.</p>
<p>Metadata</p>	<p><b>Type</b> complexvalue</p> <p><b>Properties</b> Create, Nillable, Update</p> <p><b>Description</b> The Batch Management job's metadata.</p>

Field	Details
	<p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
NamespacePrefix	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> 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 <b><i>namespacePrefix__componentName</i></b> notation.</p> <p>The namespace prefix can have one of the following values.</p> <ul style="list-style-type: none"><li>• In Developer Edition orgs, <code>NamespacePrefix</code> 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.</li><li>• In orgs that are not Developer Edition orgs, <code>NamespacePrefix</code> is set only for objects that are part of an installed managed package. All other objects have no namespace prefix.</li></ul>
ProcessGroup	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The group or team for which the Batch Management job processes records.</p>
RecordIdVariable	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The unique identifier that identifies each record that must be processed by the Batch Management job.</p>
RetryCount	<p><b>Type</b> int</p> <p><b>Properties</b> Defaulted on create, Filter, Group, Sort</p>

Field	Details
	<p><b>Description</b> Required. The number of times this Batch Management job must be rerun in case it fails.</p>
RetryInterval	<p><b>Type</b> int</p> <p><b>Properties</b> Defaulted on create, Filter, Group, Sort</p> <p><b>Description</b> Required. The number of milliseconds after which the Batch Management job must be rerun in case it fails. A retry interval can be 1,000–10,000 milliseconds.</p>
Status	<p><b>Type</b> picklist</p> <p><b>Properties</b> Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p><b>Description</b> The status of the Batch Management job. Possible values are:</p> <ul style="list-style-type: none"> <li>• Active</li> <li>• Inactive</li> </ul>
Type	<p><b>Type</b> picklist</p> <p><b>Properties</b> Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p><b>Description</b> The type of process for which the Batch Management job processes records. Possible values are:</p> <ul style="list-style-type: none"> <li>• DecisionTableRefresh</li> <li>• Flow</li> <li>• LoyaltyProgramProcess</li> </ul> <p>The default value is Flow. This field is available in API version 55.0 and later.</p>
TypeInstance	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Sort</p> <p><b>Description</b> Required. The API name of the process that the Batch Management job must execute.</p>



## Batch Management Standard Objects

Batch Management contains a standard BatchProcessJobDefinitionView object, which you can use to view all the batch jobs available in your Salesforce org, including file-based jobs.

### [BatchProcessJobDefView](#)

Represents the details of a Batch Job definition. The definition can also be file-based definitions that are available in your Salesforce org. This object is available in API version 51.0 and later.

## BatchProcessJobDefView

Represents the details of a Batch Job definition. The definition can also be file-based definitions that are available in your Salesforce org. This object is available in API version 51.0 and later.

### Supported Calls

`describeSObjects()`, `query()`

### Fields

Field	Details
DurableId	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The unique identifier for the field. Always retrieve this value before using it, as the value isn't guaranteed to stay the same from one release to the next. Simplify queries by using this field instead of making multiple queries.</p>
IsActive	<p><b>Type</b> boolean</p> <p><b>Properties</b> Defaulted on create, Filter, Group, Sort</p> <p><b>Description</b> Indicates whether the definition is active.</p>
Label	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The label of the Batch Job definition.</p>

### EDITIONS

Available in: Lightning Experience

Available in: Batch Management is available with Enterprise, Unlimited, and Performance Editions with the Loyalty Management, Manufacturing Cloud, Rebate Management, or Accounting Subledger

Field	Details
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The name of the Batch Job definition.</p>
NamespacePrefix	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> 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.</p> <p>The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"><li>• 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.</li><li>• 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.</li></ul>
ProcessDefinition	<p><b>Type</b> textarea</p> <p><b>Properties</b> Nillable</p> <p><b>Description</b> The name of the process group for the batch process job definition.</p>
ProcessGroup	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The team or group for which the definition processes records.</p>
SourceObjectName	<p><b>Type</b> string</p>

Field	Details
	<p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The API name of the object whose records are processed.</p>
Type	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The type of process for which the Batch Management job processes records. Possible values are:</p> <ul style="list-style-type: none"> <li>• Flow</li> <li>• LoyaltyProgramProcess</li> </ul> <p>This field is available in API version 55.0 and later.</p>
TypeInstance	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The API name of the process that's processed by the Batch Job definition.</p>

## Batch Management Metadata API


Use a Metadata API to create, update, and activate Batch Management jobs.

### [BatchProcessJobDefinition](#)

Represents the details of a Batch Management job definition. This type extends the Metadata metadata type and inherits its `fullName` field.

### BatchProcessJobDefinition

Represents the details of a Batch Management job definition. This type extends the Metadata metadata type and inherits its `fullName` field.

 **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.

#### EDITIONS

Available in: Lightning Experience

Available in: Batch Management is available with Enterprise, Unlimited, and Performance Editions with the Loyalty Management, Manufacturing Cloud, Rebate Management, or Accounting Subledger

## File Suffix and Directory Location

BatchProcessJobDefinition components have the suffix `.batchProcessJobDefinition` and are stored in the `batchProcessJobDefinitions` folder.

## Version

BatchProcessJobDefinition components are available in API version 51.0 and later.


## Special Access Rules

To use this metadata type, your Salesforce org must have the Loyalty Management or the Rebate Management license. The Loyalty Program Process type is only available in orgs that have Loyalty Management enabled.

## Fields

Field Name	Field Type	Description
<code>batchSize</code>	integer	Required. Number of records that each Batch Management job can process. Flow type Batch Management jobs can process up to 2000 records and Loyalty Program Process type Batch Management jobs can process up to 250 records.
<code>dataSource</code>	<a href="#">BatchDataSource</a> on page 701[]	Required. Source of information whose records must be processed by the Batch Management job.
<code>description</code>	string	Description of the Batch Management job, up to 255 characters.
<code>executionProcessApiName</code>	string	API name of process that must be executed by the Batch Management job. This field is available in API version 55.0 and later. <ul style="list-style-type: none"> <li>• If the batch job's type is Flow, enter the API name of an active flow that the batch job must execute.</li> <li>• If the batch job's type is Loyalty Program Process, enter: <ul style="list-style-type: none"> <li>– Transaction_Journals if you want the batch job to process Transaction Journal records by applying the applicable active loyalty program process of the type TransactionJournal.</li> <li>– API name of an active loyalty process of the type TierProcessing if you want the batch job to run the loyalty program process to assess the tier of eligible members. The API name consists of the name of the process, the process type, and the name of the loyalty program separated by two consecutive underscores. For example, the process API name is <code>Update Member Tier__TierProcessing__Inner Circle</code> if the process name is Update Member Tier, the process type is TierProcessing, and the loyalty program name is Inner Circle.</li> </ul> </li> </ul>

You can use database-based APEX classes that let you use flex queues in the Batch Management job, allowing to place more than 5 jobs in a queue. This functionality is applicable to all Industry Clouds that use managed packages. See [Apex Flex Queue](#).

Field Name	Field Type	Description
flowApiName	string	API name of an active flow process that must be executed by the Batch Management job.   <b>Note:</b> You can either specify the flow API name in the <code>executionProcessApiName</code> field or in the <code>flowApiName</code> field.
flowInputVariable	string	Input variable of associated flow that is used by the batch job to uniquely identify records.
masterLabel	string	Required. Name of the Batch Management job, up to 80 characters.
processGroup	string	Required. Name of the group for which the Batch Management job processes records.
retryCount	integer	Required. Number of times this Batch Management job must be rerun in case it fails. The maximum retry count is 3. Valid values are 1–3.
retryInterval	integer	Required. Number of milliseconds after which the Batch Management job must be rerun in case it fails. Valid values are 1,000–10,000.
status	string	Indicates the status of the Batch Management job. Valid values are <code>Active</code> and <code>Inactive</code> .
type	string (enumeration of type string)	The type of process that the Batch Management job must execute. This field is available in API version 55.0 and later. Valid values are: <ul style="list-style-type: none"> <li>• <code>Flow</code></li> <li>• <code>Loyalty Program Process</code></li> </ul>

## BatchDataSource

Represents the source of information whose records must be processed by the Batch Management job.

### Fields

Field Name	Field Type	Description
condition	string	Required. Criteria defined to filter the records.
criteria	string	Type of filter criteria that's used to filter records for processing.
filters	<a href="#">BatchDataSrcFilterCriteria</a> on page 702[]	Filter criterion that decides which records must be processed by the Batch Management job.
sourceObject	string	Required. API name of an object whose records must be processed by the batch job.  If the batch job type is <code>Loyalty Program Process</code> , the source object must be: <ul style="list-style-type: none"> <li>• <code>TransactionJournal</code> if the batch job is used to process transaction journals by applying the applicable loyalty program process.</li> </ul>

Field Name	Field Type	Description
		<ul style="list-style-type: none"> <li>An object that stores the details of loyalty program members whose tier must be assessed by the loyalty program process specified in the <code>executionProcessApiName</code> field.</li> </ul>
<code>sourceObjectField</code>	string	<p>API name of the source object field that uniquely identifies records for which the batch job is executed. This field is available in API version 57.0 and later.</p> <p>This field is only applicable when the batch job's type is Loyalty Program Process and a TierProcess type active loyalty program process is specified in the <code>executionProcessApiName</code> field. Specify the API name of a field that is a lookup to the LoyaltyProgramMember object and uniquely identifies the members whose tier must be assessed.</p>

## BatchDataSrcFilterCriteria

Represents the filter conditions that decide which records must be processed by the Batch Management job.

### Fields

Field Name	Field Type	Description
<code>dynamicValueType</code>	string	Data type of the input variable used as a filter.
<code>fieldName</code>	string	Required. Name of the field that must be used to filter records.
<code>fieldValue</code>	string	Required. Value of the field that must be filtered. Specify the field if <code>isDynamicValue</code> is set to <code>False</code> .
<code>isDynamicValue</code>	boolean	Required. Indicates whether the filter criteria is dynamic.
<code>operator</code>	string (enumeration of type string)	<p>Required. Operator that is specified in the filter criteria. Valid values are:</p> <ul style="list-style-type: none"> <li><code>equals</code></li> <li><code>excludes</code></li> <li><code>greaterThan</code></li> <li><code>greaterThanOrEqualTo</code></li> <li><code>in</code></li> <li><code>includes</code></li> <li><code>lessThan</code></li> <li><code>LessThanOrEqualTo</code></li> <li><code>GreaterOrEqual</code></li> <li><code>like</code></li> <li><code>notEquals</code></li> <li><code>notIn</code></li> </ul>
<code>sequenceNo</code>	integer	Required. Sequence number used to refer the criteria in a filter.

## Declarative Metadata Sample Definition

The following is an example of a BatchProcessJobDefinition component.

```
<?xml version="1.0" encoding="UTF-8"?>
<BatchProcessJobDefinition xmlns="http://soap.sforce.com/2006/04/metadata">
  <batchSize>10</batchSize>
  <dataSource>
    <condition>1</condition>
    <criteria>all</criteria>
    <filters>
      <dynamicValue>>false</dynamicValue>
      <dynamicValueType>string</dynamicValueType>
      <fieldName>Name</fieldName>
      <fieldValue>abcd</fieldValue>
      <operator>equals</operator>
      <sequenceNo>1</sequenceNo>
    </filters>
    <sourceObject>Account</sourceObject>
  </dataSource>
  <flowApiName>Flow1</flowApiName>
  <flowInputVariable>recordId</flowInputVariable>
  <masterLabel>BatchJob1</masterLabel>
  <processGroup>Loyalty</processGroup>
  <retryCount>2</retryCount>
  <retryInterval>1000</retryInterval>
  <status>Inactive</status>
  <description>test</description>
  <type>Flow</type>
  <executionProcessApiName>testFlow</executionProcessApiName>
</BatchProcessJobDefinition>
```

The following is an example of a Flow object used in Metadata API.

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
  ~ Copyright 2020 Salesforce, Inc.
  ~ All Rights Reserved
  ~ Company Confidential
-->
<Flow xmlns="http://soap.sforce.com/2006/04/metadata">
  <apiVersion>51.0</apiVersion>
  <interviewLabel>Flow1 {!$Flow.CurrentDateTime}</interviewLabel>
  <label>Flow1</label>
  <processMetadataValues>
    <name>BuilderType</name>
    <value>
      <stringValue>LightningFlowBuilder</stringValue>
    </value>
  </processMetadataValues>
  <processMetadataValues>
    <name>OriginBuilderType</name>
    <value>
      <stringValue>LightningFlowBuilder</stringValue>
    </value>
  </processMetadataValues>
</Flow>
```

```

    </value>
  </processMetadataValues>
  <processType>AutoLaunchedFlow</processType>
  <recordLookups>
    <name>getAcc</name>
    <label>getAcc</label>
    <locationX>614</locationX>
    <locationY>465</locationY>
    <assignNullValuesIfNoRecordsFound>>false</assignNullValuesIfNoRecordsFound>
    <filterLogic>and</filterLogic>
    <filters>
      <field>Id</field>
      <operator>EqualTo</operator>
      <value>
        <elementReference>recordId</elementReference>
      </value>
    </filters>
    <getFirstRecordOnly>>true</getFirstRecordOnly>
    <object>Account</object>
    <storeOutputAutomatically>>true</storeOutputAutomatically>
  </recordLookups>
  <start>
    <locationX>73</locationX>
    <locationY>213</locationY>
    <connector>
      <targetReference>getAcc</targetReference>
    </connector>
  </start>
  <status>Draft</status>
  <variables>
    <name>recordId</name>
    <dataType>String</dataType>
    <isCollection>>false</isCollection>
    <isInput>>true</isInput>
    <isOutput>>false</isOutput>
  </variables>
</Flow>

```

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>BatchProcessJobDefinition</name>
  </types>
  <types>
    <members>Flow1</members>
    <name>Flow</name>
  </types>
  <version>51.0</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](#).

## Batch Management Invocable Actions

Run an active Batch Management job definition. For more information on custom invocable actions, see [REST API Developer Guide](#) and [Actions Developer Guide](#).

### [Batch Job Actions](#)

Run an active Batch Management job definition. This action executes a defined Batch Management job asynchronously.

### [Submit Failed Records Batch Job](#)

Run to resubmit an existing batch job with failed records for processing. This action executes the batch job asynchronously.

## Batch Job Actions

Run an active Batch Management job definition. This action executes a defined Batch Management job asynchronously.

A Batch Management job processes a flow in manageable parts. For more information about running an active Batch Management jobs, see [Schedule a Batch Job](#) in Salesforce Help.

This object is available in API version 51.0 and later.

## Supported REST HTTP Methods

### URI

`/services/data/v $xx.x$ /actions/custom/batchJobAction`

### Formats

JSON

### HTTP Methods

GET, POST

### Authentication

Authorization: Bearer *token*

## Inputs

The batch job action uses the batch job definition ID and input variables to execute the job and generate a batch job ID. The input values vary according to the input variables in that flow.

## Outputs

Output	Details
<code>batchJobId</code>	<b>Type</b> string

**Output****Details****Description**

ID of the batch job generated after the request is successful. This batch job ID is used to track the progress of the batch job in Monitor Workflow Services.

**Usage****JSON Sample Request**

```
{
  "noOfEmployees" : 900,
  "accountIndustry" : "Technology"
}
```

**JSON Sample Response**

```
{
  "batchJobId": "01Mxx0000A000001EAA"
  "accepted": "true"
}
```

**Submit Failed Records Batch Job**

Run to resubmit an existing batch job with failed records for processing. This action executes the batch job asynchronously.

This object is available in API version 52.0 and later.

**Supported REST HTTP Methods****URI**

/services/data/v**XX.X**/actions/standard/submitFailedRecordsBatchJob

**Formats**

JSON

**HTTP Methods**

GET, POST

**Authentication**

Authorization: Bearer *token*

**Inputs****Input****Details**

failedRecordIds

**Type**

array

**Description**

The IDs of failed records in a batch job.

Input	Details
parentBatchJobId	<b>Type</b> string <b>Description</b> Required. The ID of a batch job with failed records.

## Outputs

Output	Details
batchJobId	<b>Type</b> string <b>Description</b> The ID of a batch job generated after the request is successful. This batch job ID is used to track the progress of the batch job in Monitor Workflow Services in the org.
status	<b>Type</b> string <b>Description</b> Indicates whether a batch job succeeded or failed.

## Usage

### JSON Sample Request

```
{
  "inputs": [{
    "parentBatchJobId": "0mdRM0000004DXrYAM",
    "failedRecordIds": [
      "001RM000005AG0byAG", "001RM000005AERZYA4", "001RM000005AG0WYAW"
    ]
  }]
}
```

### JSON Sample Response

```
[ {
  "actionName" : "submitFailedRecordsBatchJob",
  "errors" : null,
  "isSuccess" : true,
  "outputValues" : {
    "batchJobId" : "0mdRM0000004DZ9YAM"
  }
} ]
```

# Monitor Workflow Services

The Monitor Workflow Services standard objects can be used to track the run of Data Processing Engine definitions and Batch Management jobs. During a run, you can view details about each part that the run is broken down into. After the run is complete, you can view its status and the records which weren't processed during the run.

The objects of Monitor Workflow Services aren't available in Object Manager of your Salesforce org.

### [BatchJob](#)

Represents an instance of a batch job that is either running and has been run. This object is available in API version 51.0 and later.

### [BatchJobPart](#)

Represents one part of a batch job. This object is available in API version 51.0 and later.

### [BatchJobPartFailedRecord](#)

Represents records that a batch job part couldn't successfully process. This object is available in API version 51.0 and later.

## EDITIONS

Available in: Lightning Experience

Available in: Monitor Workflow Services is available with Enterprise, Unlimited, and Performance Editions where Data Processing Engine or Batch Management is available

## BatchJob

Represents an instance of a batch job that is either running and has been run. This object is available in API version 51.0 and later.

## Supported Calls

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

## Fields

Field	Details
<code>BatchJobDefinitionId</code>	<p><b>Type</b> reference</p> <p><b>Properties</b> Filter, Group, Sort</p> <p><b>Description</b> The unique identifier of the associated batch job definition. This is a relationship field.</p> <p><b>Relationship Name</b> BatchJobDefinition</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> BatchJobDefinition</p>

Field	Details
BatchJobDefinitionName	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Sort</p> <p><b>Description</b> The developer name of the associated batch job definition.</p>
EndTime	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The timestamp when the batch job run was completed.</p>
ErrorDescription	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The error message in case the batch job run failed.</p>
ExternalReference	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The unique identifier of the process that's running or has run the batch job.</p>
IsDebugOn	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> The default value is <code>false</code>.</p>
IsDebugRecipeDeleted	<p><b>Type</b> boolean</p> <p><b>Properties</b> Create, Defaulted on create, Filter, Group, Sort, Update</p> <p><b>Description</b> The default value is <code>false</code>.</p>

Field	Details
LastReferencedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The timestamp for when the current user last viewed the batch job.</p>
LastViewedDate	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The timestamp for when the current user last viewed this item.</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, idLookup, Sort</p> <p><b>Description</b> The name of the batch job.</p>
OwnerId	<p><b>Type</b> reference</p> <p><b>Properties</b> Filter, Group, Sort</p> <p><b>Description</b> Unique identifier of the user who initiated the batch job run. This is a polymorphic relationship field.</p> <p><b>Relationship Name</b> Owner</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> Group, User</p>
ProcessGroup	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Sort</p> <p><b>Description</b> The group or team for which the batch job is run.</p>

Field	Details
RetryCount	<p><b>Type</b> int</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The number of times the batch job run is automatically rerun in case it fails.</p>
RuntimeParameter	<p><b>Type</b> textarea</p> <p><b>Properties</b> Nillable</p> <p><b>Description</b> The values of the input variables that are used as filter criteria in a Batch Management job.</p>
StartTime	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The timestamp when the batch job run was started.</p>
Status	<p><b>Type</b> picklist</p> <p><b>Properties</b> Filter, Group, Restricted picklist, Sort</p> <p><b>Description</b> The status of the batch job run. Possible values are:</p> <ul style="list-style-type: none"> <li>• Canceled</li> <li>• Completed</li> <li>• CompletedWithFailures</li> <li>• Failed</li> <li>• InProgress</li> <li>• Submitted</li> </ul>
Type	<p><b>Type</b> picklist</p> <p><b>Properties</b> Defaulted on create, Filter, Group, Restricted picklist, Sort</p> <p><b>Description</b> The type of batch job that is run.</p>

Field	Details
	<p>Possible values are:</p> <ul style="list-style-type: none"> <li>• Calc—Data Processing Engine</li> <li>• DecisionTableRefresh</li> <li>• Flow</li> </ul>

## 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.

### BatchJobFeed

Feed tracking is available for the object.

### BatchJobHistory

History is available for tracked fields of the object.

## BatchJobPart

Represents one part of a batch job. This object is available in API version 51.0 and later.

When a batch job is run, it is divided in to multiple parts. Each part is used to process a specific number of records.

## Supported Calls

`describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

## Fields

Field	Details
BatchJobId	<p><b>Type</b> reference</p> <p><b>Properties</b> Filter, Group, Sort</p> <p><b>Description</b> The unique identifier of the associated batch job. This is a relationship field.</p> <p><b>Relationship Name</b> BatchJob</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> BatchJob</p>



Field	Details
EndTime	<p><b>Type</b> dateTime</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The timestamp when the batch job part was processed.</p>
ErrorDescription	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The error message in case the batch job part failed.</p>
FailedRecordCount	<p><b>Type</b> int</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The number of records that the batch job part couldn't process.</p>
FailedRecFile	<p><b>Type</b> FILE</p> <p><b>Properties</b> Filter, Group, Sort</p> <p><b>Description</b> The file containing failed records downloaded as CSV, JSON, or TXT formats during a Data Processing Engine definition's run by using CRM Analytics (CRMA) or Data Cloud runtime. Available in API version 63.0 and later.</p>
InputRecordCount	<p><b>Type</b> int</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The number of records that the batch job part must process.</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, idLookup, Sort</p>

Field	Details
	<b>Description</b> The name of the batch job part.
OutputRecordCount	<b>Type</b> int <b>Properties</b> Filter, Group, Nillable, Sort <b>Description</b> The number of records the batch job part has processed.
ParentBatchJobPartId	<b>Type</b> reference <b>Properties</b> Filter, Group, Nillable, Sort <b>Description</b> The unique identifier of the part batch job part associated with the batch job part. This is a relationship field. <b>Relationship Name</b> ParentBatchJobPart <b>Relationship Type</b> Lookup <b>Refers To</b> BatchJobPart
RetryCount	<b>Type</b> int <b>Properties</b> Filter, Group, Nillable, Sort <b>Description</b> The number of times the batch job part is automatically rerun in case it fails.
StartTime	<b>Type</b> dateTime <b>Properties</b> Filter, Nillable, Sort <b>Description</b> The timestamp when the batch job part's run was started.
Status	<b>Type</b> picklist

Field	Details
Type	<p><b>Properties</b> Defaulted on create, Filter, Group, Restricted picklist, Sort</p> <p><b>Description</b> The status of the batch job part. Possible values are:</p> <ul style="list-style-type: none"> <li>• Canceled—Canceled</li> <li>• Completed</li> <li>• Failed</li> <li>• InProgress—In Progress</li> </ul> <hr/> <p><b>Type</b> picklist</p> <p><b>Properties</b> Filter, Group, Nillable, Restricted picklist, Sort</p> <p><b>Description</b> The type of node in case the associated batch job is of the type Calc (Data Processing Engine). Possible values are:</p> <ul style="list-style-type: none"> <li>• Aggregate</li> <li>• Append</li> <li>• Compute</li> <li>• Custom</li> <li>• Datasync</li> <li>• Filter</li> <li>• Forecast</li> <li>• Hierarchy</li> <li>• Join</li> <li>• Slice</li> <li>• Source</li> <li>• Transform</li> <li>• Writeback</li> </ul>

## 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.

### BatchJobPartFeed

Feed tracking is available for the object.

### BatchJobPartHistory

History is available for tracked fields of the object.

## BatchJobPartFailedRecord

Represents records that a batch job part couldn't successfully process. This object is available in API version 51.0 and later.

### Supported Calls

`describeLayout()`, `describeSObjects()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`

### Fields

Field	Details
<code>BatchJobId</code>	<p><b>Type</b> reference</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The unique identifier of the associated batch job. This is a relationship field.</p> <p><b>Relationship Name</b> BatchJob</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> BatchJob</p>
<code>BatchJobPartId</code>	<p><b>Type</b> reference</p> <p><b>Properties</b> Filter, Group, Sort</p> <p><b>Description</b> The unique identifier of the associated batch job part. This is a relationship field.</p> <p><b>Relationship Name</b> BatchJobPart</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> BatchJobPart</p>
<code>ErrorDescription</code>	<p><b>Type</b> string</p>

Field	Details
	<p><b>Properties</b> Filter, Nillable, Sort</p> <p><b>Description</b> The error message that indicates why the batch job part couldn't process the records.</p>
Name	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, idLookup, Sort</p> <p><b>Description</b> The name of the batch job part failed record.</p>
Record	<p><b>Type</b> string</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The unique identifier of the batch record that processed the failed records.</p>
ResubmittedBatchJobId	<p><b>Type</b> reference</p> <p><b>Properties</b> Filter, Group, Nillable, Sort</p> <p><b>Description</b> The batch job used to submit failed records. This object is available in API version 52.0 and later.  This is a relationship field.</p> <p><b>Relationship Name</b> ResubmittedBatchJob</p> <p><b>Relationship Type</b> Lookup</p> <p><b>Refers To</b> BatchJob</p>
Status	<p><b>Type</b> picklist</p> <p><b>Properties</b> Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p><b>Description</b> Specifies the status of the failed records. This object is available in API version 52.0 and later.  Possible values are:</p>

Field	Details
	<ul style="list-style-type: none"><li>Failed</li><li>Resubmitted</li></ul> <p>The default value is 'Failed'.</p>

## 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.

### **BatchJobPartFailedRecordFeed**

Feed tracking is available for the object.

### **BatchJobPartFailedRecordHistory**

History is available for tracked fields of the object.