Actions Developer Guide

Developer Guide

Version 63.0, Spring '25

Last updated: February 14, 2025

CONTENTS

CHAPTER 1 Introducing Actions

Use actions to add more functionality to your applications. Choose from standard actions, such as posting to Chatter or sending email, or create actions based on your company's needs.

For example, you can:

- Add features and functionality to your existing Lightning Platform tools.
- Build dynamic modules for Lightning Platform development into your enterprise integration tools.
- Build specialized development tools for a specific application or service.

You can batch actions to improve performance in API version 35.0 and later.

Overview

Actions allow you to build custom development tools for Lightning Platform applications.

Actions are about "getting things done" in Salesforce. They encapsulate a piece of logic that allows a user to perform some work, such as sending email. When an action runs, it saves changes in your organization by updating the database.

Actions are easy to discover and use, and also easy to understand and implement. Every button and link in Salesforce can be considered an action. A consistent Actions API and framework support the creation and distributed use of actions throughout Salesforce. Actions are available in the REST API.

The types of actions are:

Type	Description
InvocableAction	Invocable actions can be invoked from a common endpoint in the REST API. They provide "describe" support – a programmatic mechanism to learn about all invocable actions on the platform. There are two types of invocable actions. **Standard action** A standard action is ready to use right away. The work it performs is predefined, along with its inputs and outputs, and they're available in every organization. **Custom action** You create custom actions because these actions require a definition. For example, to use an Apex action, create the Apex class method for the action.
QuickAction	Quick Actions, formerly known as Publisher Actions, use page layouts to make it easy for administrators to configure an action to create or update a record. The API always works with an sObject.
StandardButton	Standard buttons are URLs allowing users to either go to another page (for example, the Edit page) or accomplish some task (for example, lead conversion).
CustomButton	Custom buttons are URLs that an administrator can specify and when included on a page and clicked, will redirect a user to that URL.

To call an action from a flow, use `FlowActionCall`, as described in the *Metadata API Developer's Guide*.

The `If-Modified-Since` header can be used with actions, with a date format of `EEE, dd MMM yyyy HH:mm:ss z`. When this header is used, if the action metadata has not changed since the provided date, a `304 Not Modified` status code is returned, with no response body.

SEE ALSO:

REST API Developer Guide : Invocable Actions

Invoking Actions

Most actions are invoked using the same JSON body format. The top-level JSON key name must be `inputs`.

Note: Invoke Salesforce Order Management actions with the corresponding Connect REST API resources or Apex ConnectApi methods, not the standard endpoints.

The following example request shows two Chatter posts made with a single Post to Chatter action.

```
POST /services/data/vXX.X/actions/standard/chatterPost

{ "inputs" :
  [
  {
    "subjectNameOrId" : "jsmith@salesforce.com",
    "type" : "user",
    "text" : "first chatter post!"
  },
  {
    "subjectNameOrId" : "hsmith@salesforce.com",
    "type" : "user",
    "text" : "second chatter post!"
  }
  ]
}
```

Here is the response.

```
[ {
  "actionName" : "chatterPost",
  "errors" : null,
  "isSuccess" : true,
  "outputValues" : {
    "feedItemId" : "0D5D0000000kynqKBA"
  }
}, {
  "actionName" : "chatterPost",
  "errors" : null,
  "isSuccess" : true,
  "outputValues" : {
    "feedItemId" : "0D5D0000000kynrKBz"
  }
} ]
```

Standard actions return their name in `actionName`. The value of `actionName` varies for custom actions.

Action	`actionName` **value**
Flow	The flow name
Apex	The class's invocable method name
Quick action	`<object name>.<quick action name>` For a global quick action, there's no `<object name>.` prefix.
Email alert	`<object name>.<email alert name>`
Send notification	The API name of the notification type
Generate Prompt Response	The API name of the prompt template

SEE ALSO:

REST API Developer Guide : Invocable Actions

Available Actions

The available actions are:

Action	Description
Apex Actions	Invoke Apex methods annotated with `@InvocableMethod` and include custom parameters with `@InvocableVariable`.
Assign Candidates to Research Study Group Action	Randomly assign candidates, enrolled in the clinical trials, to research study comparison groups.
Asset Lifecycle Actions	Create or update an asset from an order or order item. Additionally, initiate the amendment, cancellation, or renewal of an asset.
Assign Enablement Program	Automatically assign a user to an Enablement program.
Apply Case Classification Recommendations	Applies Einstein's recommended values for fields on a given case record, and returns the updated case record.
Batch Management Actions	Manage your Batch Management jobs by using invocable actions.
Billing Actions	Manage billing operations by using invocable actions.
B2B and D2C Commerce Actions	Manage B2B Commerce integrations and store checkout flow. Record and reverse tax transactions in external systems like Stripe based on order changes, and create subscription records for orders with subscription products.
Commerce Checkout Flow Actions	Manage Commerce integrations and store checkout flow.
Create Service Report Actions	Creates a service report for a service appointment, work order, or work order line item.
Custom Notification Actions	Send custom notifications to recipients via desktop or mobile channels.
Decision Table Actions	Invokes a decision table or refreshes business rules for an active decision table.

Action	Description
Data Processing Engine Actions	Runs an active Data Processing Engine definition.
Deploy Data Kit Components Action	Deploys data kit components of a package to a target org via a flow action.
Dynamic Revenue Orchestrator Actions	Submit an order or a sales transaction to Dynamic Revenue Orchestrator (DRO) for fulfillment.
Email Alert Actions	Send emails from flows by reusing already-configured workflow email alerts.
Einstein Bots Actions	Search for knowledge articles based on data category and data category groups.
Einstein Visit Recommendation Action	Save visit and task recommendation decisions.
Initiate Natural Language Processing Action	Create a record for the AI natural language processing result and initiate text processing by using the service specified in the related record.
Financial Services Cloud Actions	Create person accounts, financial accounts, and related records from a residential loan application for Financial Services Cloud.
Flow Actions	Invoke an active autolaunched flow or active invocable process that exists in the current org.
Fundraising for Nonprofit Cloud Actions	Manage gift commitments, gift commitment schedules, gift default schedules, gift transaction designations, and gift entries for Fundraising.
Generate Order Summary Action	Generate a URL so that authenticated and guest users can access order details.
Generate Research Study Block Action	Generate research study randomization block records to link each block with a specific research study comparison group by using the randomization process.
Media Integration Procedure Action	Call an Integration Procedure from a Salesforce Flow to process media content.
Prompt Template Actions	Generate a response based on the large language model (LLM) response for the specified prompt template and inputs.
Generate Work Orders Actions	Generates work orders from a maintenance plan.
Health Cloud Actions	Automate healthcare-related tasks using invocable actions.
Quote and Order Capture Actions	Create an order from a quote record.
Sales Engagement Actions	Manage cadence targets by using invocable actions.
Knowledge Actions	Manage your Knowledge articles by using invocable actions.
Live Message Notification Actions	Use messaging templates to send notifications to users over communication channels, such as SMS, WhatsApp, and Facebook Messenger, when certain conditions are met.
Loyalty Management Actions	Create and manage loyalty programs for your organization by using the standard and custom invocable actions.
Manufacturing Cloud Actions	Automate business processes related to account forecast, sales agreements, and account manager target values.
Net Zero Cloud Actions	Track and manage environmental impact for precise calculation and analysis of carbon emissions.

Action	Description
Omni-Channel Action	Create a `PendingServiceRouting` record used for Omni-Channel skills-based routing.
Apply Payment Action	Applies a payment record to an invoice header by creating a PaymentLineInvoice record with a type of Applied.
Payment Sale Action	Capture a payment without any prior authorization and create a payment record. The payment sale transaction consists of an authorize request and a capture request made to the payment gateway at the same time. This way, the merchant can request funds to be transferred to the merchant account in a single command, with no further action required.
PlatformAction	PlatformAction is a virtual read-only object. It enables you to query for actions displayed in the UI, given a user, a context, device format, and a record ID. Examples include standard and custom buttons, quick actions, and productivity actions.
Preview Cart to Exchange Order	Generate preview details of an exchange order for specified order summary, exchange cart ID, and reference record ID.
Post to Chatter Actions	Post a message to a specified feed, such as to a Chatter group or a case record. The message can contain mentions and topics, but only text posts are supported.
Public Sector Solutions Actions	Create a benefit disbursement for an eligible benefit assignment or run a Data Processing Engine definition to process an asynchronous batch job.
Quick Actions	Use a quick action to create a task or a case. Invoke existing quick actions, both global and object-specific, to create records, update records, or log calls.
Rebate Management Actions	Create and manage rebate programs and manage payouts and transactions by using the Rebate Management invocable actions.
Referral Marketing Actions	Create and manage referral programs for your organization.
Refresh Metric Actions	Update a metric's Current Value field if it's linked to a summary field in a Salesforce report. The refresh runs as the metric owner.
Salesforce Omnichannel Inventory Actions	Manage inventory availability and provide omnichannel commerce experiences in flows with Salesforce Omnichannel Inventory.
Salesforce Order Management Actions	Manage, fulfill, and service orders in flows with Salesforce Order Management.
Salesforce Pricing Actions	Invoke the Pricing Connect API by providing the context, pricing procedure, and price waterfall details. Additionally, you can also specify the pricing data and details of a context to invoke the Pricing Connect API.
Schedule Group Visits Actions	
Send Notification Actions	Call a notification type to send. Each Send Notification action corresponds to an available notification type.
Session-Based Permission Set Actions	Activate or deactivate a session-based permission set for the current user's API session.
Simple Email Actions	Send an email where you specify the subject, body, and recipients.

Action	Description
Submit for Approval Actions	Submit a Salesforce record for approval if an approval process is defined for the current entity.
Submit Exchange Order	Submits an exchange order based on the specified information.
Survey Invitation Actions	Send email survey invitations to leads, contacts, and users in your org based on an action. Also, send customized notifications to users about important events or updates to the records that they're working on.
Usage Management Actions	Manage consumption of usage-based products by using the invocable actions.

CHAPTER 2 Action Objects

This is the reference for quick actions and dynamic actions. Invocable actions are also known as dynamic actions.

Apex Actions

Invoke Apex methods annotated with `@InvocableMethod` and include custom parameters with `@InvocableVariable`.

This object is available in API version 33.0 and later.

Supported REST HTTP Methods

URI

Get a list of available Apex actions:

`/services/data/vXX.X/actions/custom/apex`

Get information about a specific Apex action:

`/services/data/vXX.X/actions/custom/apex/action_name`

Formats

JSON, XML

HTTP Methods

GET, HEAD, POST

Authentication

`Authorization: Bearer token`

Parameters

None

Example

This example invokes the Apex action called `ActionTestWithSObject`, which takes a list of Accounts, increases the employee count for each account by one, and returns an updated list of account IDs. The top-level key name in the JSON request body must be `inputs`.

```
POST /services/data/vXX.X/actions/custom/apex/ActionTestWithSObject

{

 "inputs": [
  {
   "objects": {
    "attributes" : {
     "type" : "Account"
    },
    "Name": "Acme"
   }
  },
  {
```

```
    "objects": {
     "attributes" : {
      "type" : "Account"
     },
     "Name": "Global Media"
    }
   }
  ]
 }
```

Here's the Apex code.

```
public class ActionTestWithSObject {

 @InvocableMethod(label='Action Test With sObject' description='Given a list of sObjects
 (Accounts), increase employee count by one and return account ID' category='Account')

 public static List<String> getAccountNames(List<Account> objects) {

  List<String> accountIds = new List<String>();

  for (Account a : objects) {
   Account retrievedAccount = [SELECT Id, NumberOfEmployees FROM Account WHERE Name
=:a.Name LIMIT 1];
   retrievedAccount.NumberOfEmployees += 1;
   update retrievedAccount;
   accountIds.add(retrievedAccount.Id);
  }

  return accountIds;
 }
}
```

Note: The resource is the name of the Apex class, not the Apex method. In this example, the resource is `/ActionTestWithSObject`, not `/getAccountNames`.

Notes

- Describe and invoke for an Apex action respect the profile access for the Apex class. If you don't have access, an error is issued.
- If you add an Apex action to a flow, and then remove the `@InvocableMethod` annotation from the Apex class, you get a runtime error in the flow.
- If an Apex action is used in a flow, packageable components that reference these elements aren't automatically included in the package. For example, if you use an email alert, you must manually add the email template that is used by that email alert. To deploy the package successfully, manually add those referenced components to the package.
- An Apex invocable action can be declared `public` or `global` in a managed package. However, that action doesn't appear in Flow Builder's list of available Apex actions. Flows within the same managed package can still refer to these invocable actions. Global Apex invocable actions in a managed package can be used in flows outside the managed package, anywhere in the organization, and appear in Flow Builder's list of available Apex actions.

Inputs

Supply input values that correspond to the Apex action.

- A POST request body must use the JSON format specified in Invoking Actions.

- Apex methods annotated with `@InvocableMethod` must take a List as an input and return a List or `Null`. For more information, see `@InvocableMethod` Annotation in the *Apex Developer Guide*.
- Only the following primitive types are supported as inputs in a POST request body:
  - `Blob`
  - `Boolean`
  - `Date`
  - `Datetime`
  - `Decimal`
  - `Double`
  - `ID`
  - `Integer`
  - `Long`
  - `String`
  - `Time`
- Concrete types inherited from the sObject. In the previous example, the inherited concrete type is Account.
- A user-defined type, containing variables of the supported types and with the `InvocableVariable` annotation. To implement your data type, create a custom global or public Apex class. The class must contain at least one member variable with the invocable variable annotation.

Outputs

The Apex InvocableMethod determines the output values.

SEE ALSO:

Flow Actions

Apex Developer Guide: InvocableMethod Annotation

REST API Developer Guide : Invocable Actions

Assign Enablement Program

Automatically assign a user to an Enablement program based on your determined criteria.

To assign users to an Enablement program, enable the Design and Deliver Enablement Programs user permission.

This object is available in API version 58.0 and later.

Supported REST HTTP Methods

URI

`/services/data/v58.0/actions/standard/assignEnablementProgram`

Formats

JSON

HTTP Methods

POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`assigneeId`	**Type** ID **Description** Required. The ID of the Enablement user to assign to the program.
`notificationUserId`	**Type** ID **Description** Required. The ID of another Salesforce user to notify when the program assignment is complete. By default, a notification is sent to the user who runs this invocable action. Use `notificationUserId` to specify another user that you want to notify.
`programId`	**Type** ID **Description** Required. The ID of the program being assigned.
`startDate`	**Type** ID **Description** Required. The date that assignees can access the program. Dates for Saturdays and Sundays are automatically set to the following Monday.

Outputs

None

SEE ALSO:

Salesforce Help: Automating Enablement Program Assignment

Apply Case Classification Recommendations

Recommends values for fields on a given case record. Requires an active Einstein Case Classification model.

These actions are available in API version 55.0 and later.

Supported REST HTTP Methods

URI

Get a case SObject with recommended values for fields:

`/services/data/vXX.X/actions/standard/applyCaseClassificationRecommendations`

Formats

JSON, XML

HTTP Methods

GET, HEAD, POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`caseId`	**Type** ID **Description** Required. The ID of a case.

Outputs

Input	Details
`caseSObject`	**Type** SObject **Description** A case SObject with recommendations applied.

B2B and D2C Commerce Actions

Manage B2B Commerce integrations and store checkout flow. Record and reverse tax transactions in external systems like Stripe based on order changes, and create subscription records for orders with subscription products. For more information on standard invocable actions, see **REST API Developer Guide** and **Actions Developer Guide**.

Commerce Checkout Flow Actions

Manage your B2B Commerce integrations and create a custom checkout with Checkout Flow actions.

For more information about using Commerce Checkout Flow actions in flows, see B2B Commerce Checkout Flow Core Actions in Salesforce Help.

These actions are available in API version 50.0 and later.

Your org must have B2B Commerce enabled.

Supported REST HTTP Methods

URI

Get a specific B2B Commerce Checkout Flow action:

`/services/data/vXX.X/actions/standard/checkout_flow_action_name`

Formats

JSON, XML

HTTP Methods

GET

Authentication

`Authorization: Bearer token`

Notes

You can also call the corresponding Connect REST API endpoints or Apex ConnectApi methods. For more information, see B2B and B2B2C Commerce Resources in the *Connect REST API Developer Guide* and ConnectApi Namespace in the *Apex Developer Guide*.

Create Subscription Records Action

Creates subscription records for orders containing subscription products. It accepts either the order summary ID or the order item summary IDs as input, filters the subscription products, and creates records to manage them effectively.

Special Access Rules

This action is available in API version 63.0 and later for users with system administrator access or the Assetize Order permission set assigned, along with any of the following licenses enabled:

- B2B Commerce, or D2C Commerce and Revenue Cloud Subscription Management. When this license is enabled in your org, both `orderSummaryId` and `orderItemSummaryIds` are supported.
- B2B Commerce, or D2C Commerce and Revenue Lifecycle Management. When this license is enabled in your org, only `orderSummaryId` is supported.

Supported REST HTTP Methods

URI

`/services/data/vXX.X/actions/standard/createSubscriptionRecords`

Formats

JSON, XML

HTTP Methods

POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`webStoreId`	**Type** String **Description** Required. The ID of the web store associated with the order.
`orderSummaryId`	**Type** String **Description** The ID of the order summary record. This field is optional if `orderItemSummaryIds` is specified.
`orderItemSummaryIds`	**Type** String **Description** The list of the order item summary IDs. This field is optional if `orderSummaryId` is specified.

Outputs

Output	Details
`Subscription ProcessedItems`	**Type** STRING **Description** An array of order item summary IDs that are processed and have an associated subscription record created.

Example

Sample Request

Here's a sample request with `orderSummaryID` and `webStoreID`:

```
{
  "inputs": [
    {
      "createSubscriptionRecords": {
        "orderSummaryId": "1OsJ3000000Gmd3KAC",
        "webStoreId": "0ZE5i000000PbfKGAS"
      }
```

```
    }
  ]
}
```

Here's a sample request with `orderItemSummaryIDs` and `webStoreID`:


```
{
  "inputs": [
    {
      "createSubscriptionRecords": {
        "orderItemSummaryIds": [
          "10uJ3000000GmczIAC", "10uJ3000000GmcyIAC"
        ],
        "webStoreId": "0ZE5i000000PbfKGAS"
      }
    }
  ]
}
```


Sample Response


```
[
  {
    "actionName": "createSubscriptionRecords",
    "errors": null,
    "isSuccess": true,
    "outputValues": {
      "SubscriptionProcessedItems": [
        "10uJ3000000GmczIAC",
        "10uJ3000000GmcyIAC"
      ]
    }
  }
]
```


Process First Payment Billing for Subscriptions Action

Creates invoices for orders containing subscription products. When an order containing subscription products is placed in a B2B or D2C store, the first term payment for subscription products is captured, but an invoice isn't generated. This action uses Revenue Cloud's billing system to generate and settle a subscripition order's first-term invoice and create a billing schedule for the subscription order.

Special Access Rules

This action is available in API version 63.0 and later for users with system administrator access or the BillingTransactionToBSApiUser, InvoiceOrErrorRecoveryAPI, and RLMBilliingAccess users permissions assigned, along with any of these licenses enabled:

- B2B Commerce or D2C Commerce and Salesforce Payments
- Revenue Lifecycle Management and Salesforce Pricing

Supported REST HTTP Methods

URI

/services/data/v***XX.X***/actions/standard/processFirstPaymentBilling

Formats

JSON, XML

HTTP Methods

GET, HEAD, POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`correlationId`	**Type** String **Description** The request correlation ID used for tracking the request.
`orderSummaryId`	**Type** String **Description** The ID of the order summary record.

Outputs

Output	Details
`correlationId`	**Type** STRING **Description** The request correlationID used for tracking the request.
`requestId`	**Type** STRING **Description** The ID of the request.

Example

Sample Request

Here's a sample request with `correlationId` and `orderSummaryId`:

```
{
  "inputs": [
    {
```

```
      "orderSummaryId": "1OsLT0000098gcu0AA",
      "correlationId": "4869bd78-3af4-40ff-8475-899d711b5db4"
    }
  ]
}
```

Sample Response

```
[
  {
    "actionName": "processFirstPaymentBilling",
    "errors": null,
    "invocationId": null,
    "isSuccess": true,
    "outcome": null,
    "outputValues": {
      "requestId": "fcec4cff-3f26-45fd-8193-0a599dbb7d03",
      "correlationId": "4869bd78-3af4-40ff-8475-899d711b5db4"
    },
    "sortOrder": -1,
    "version": 1
  }
]
```

Record Tax Reversal Action

Reverses the recorded tax transactions in an external system, such as Stripe, after an order is returned or canceled.

This action is available in API version 62.0 and later for users with these licenses:

- B2B Commerce, or D2C Commerce
- Salesforce Order Management

Only store administrators can access this action.

Supported REST HTTP Methods

URI

`/services/data/vXX.X/actions/standard/recordTaxReversal`

Formats

JSON, XML

HTTP Methods

POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
taxReversalInfos	**Type** Apex-defined **Description** Required. A list of Apex `commercestoretax__TaxReversalInfo` records containing details about the tax transactions for the order that was returned or canceled.

Outputs

Output	Details
taxReversalResult	**Type** Apex-defined **Description** An Apex `commercestoretax__TaxReversalResult` record containing details about each tax transaction in the reversal request. The details include whether each reversal was successful and, if not, any error messages returned.

Example

Sample Request

```
{
   "inputs": [
      {
         "taxReversalInfos": {
            "taxReversalInfoList": [
               {
                  "orderItemSummaryId": "10uxx0000004EdmAAE",
                  "quantity": 1,
                  "amount": -10,
                  "taxAmount": 1.0
               }
            ]
         }
      }
   ]
}
```

Sample Response

```
[
   {
      "actionName": "RECORD_TAX_REVERSAL",
```

```
      "errors": null,
      "invocationId": null,
      "isSuccess": true,
      "outputValues": {
         "taxReversalResult": {
            "success": true,
            "resultItems": [
               {
                  "transactionReferenceNumber": "tax_1PcknrITDqIkouLURfR4pNAM",
                  "success": true,
                  "orderItemSummaryId": "10uxx0000004EicAAE",
                  "lineItemReference": tax_li_QZM99en1lXKf9s,
                  "errorMessage": null
               }
            ],
            "errorMessage": null
         }
      },
      "sortOrder": -1,
      "version": 1
   }
]
```


Record Tax Transaction Action

Records tax transactions from an order summary to an external system such as Stripe.

This action is available in API version 62.0 and later with these licenses:

- B2B Commerce, or D2C Commerce
- Salesforce Order Management

Only store administrators can access this action.

Supported REST HTTP Methods

URI

`/services/data/vXX.X/actions/standard/recordTaxTransaction`

Formats

JSON, XML

HTTP Methods

POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`orderSummaryId`	**Type** string **Description** Required. The ID of the order summary to record tax transaction for.

Outputs

Output	Details
`taxTransactionResult`	**Type** Apex-defined **Description** An `Apex commercestoretax__TaxTransactionResult` record that contains details about the tax transactions recorded.

Example

Sample Request

```
{
   "inputs": [
      {
         "orderSummaryId": "1Osxx0000004CVcCAM"
      }
   ]
}
```

Sample Response

```
[
  {
    "actionName": "RECORD_TAX_TRANSACTION",
    "errors": null,
    "invocationId": null,
    "isSuccess": true,
    "outputValues": {
      "taxTransactionResult": {
        "success": true,
        "resultItems": [
          {
            "transactionReferenceNumber": "tax_1PcknrITDqIkouLURfR4pNAM",
            "success": true,
            "orderItemSummaryId": "10uxx0000004EicAAE",
```

```
            "errorMessage": null,
            "calculationReferenceNumber": null
          },
          {
            "transactionReferenceNumber": "tax_1PcknrITDqIkouLURfR4pNAM",
            "success": true,
            "oderItemSummaryId": "10uxx0000004EibAAE",
            "errorMessage": null,
            "calculationReferenceNumber": null
          }
        ],
        "orderSummaryId": "1Osxx0000004CVc",
        "errorMessage": null
      }
    },
    "sortOrder": -1,
    "version": 1
  }
]
```


Commerce Checkout Flow Actions

Manage your Commerce integrations and create a custom checkout with Checkout Flow actions.

For more information about using Commerce Checkout Flow actions in flows, see Commerce Checkout Flow Core Actions in Salesforce Help.

These actions are available in API version 55.0 and later.

Supported REST HTTP Methods

URI

Get a specific Commerce Checkout Flow action:

`/services/data/vXX.X/actions/standard/flow_action_name`

Formats

JSON, XML

HTTP Methods

GET

Authentication

`Authorization: Bearer token`

Notes

You can also call the corresponding Connect REST API endpoints or Apex ConnectApi methods. For more information, see B2B and B2B2C Commerce Resources in the *Connect REST API Developer Guide* and ConnectApi Namespace in the *Apex Developer Guide*.

Create Service Document Actions

Create service documents from work orders, work order line items, or service appointments.

This object is available in API version 60.0 and later.

Supported REST HTTP Methods

URI

`/services/data/v60.0/actions/standard/createServiceDocument`

Formats

JSON

HTTP Methods

POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
recordId	**Type** string **Description** Required. The record ID of a work order, work order line item, or service appointment used to generate the service document. Create a Lightning web component to use a Custom Property Editor (CPE) to validate the `recordId` to avoid deployment issues.
templateId	**Type** string **Description** Required, if the `recordId` is a work order, work order line item, or service appointment. The ID of the service document template to use when generating the document.
locale	**Type** string **Description** Optional. Specifies the language for service document localization. The default is the user's language. Used when generating a document in a different language from the user's language. See a list of supported languages in Supported Languages. You can only input language for locale. For example, use `es` for Spanish. Using language and country, for example `es_ES` for Spanish associated with Spain, results in error.
title	**Type** string **Description** Optional. The value used to name the document that's generated and saved.
documentType	**Type** string

Input	Details
	Description Optional. Value that allows users to generate different types of documents by using the service. Valid values are: • `ServiceDocument`—Type of service document, such as service agreement or service contract. • `QuoteDocument`—Type of quote document, such as sales quote or service quote. • `SfsQuoteDocument`—Type of quote document for Salesforce Field Service (SFS), suitable for mobile use. This document is stored in the QuoteDocument object, and is generated through flow-based processes that link to related service documents. The default value is `ServiceDocument`.
`pdfReportId`	**Type** string **Description** Optional value corresponding to `recordId` and `templateId`. However, the value is required if you aren't generating the document from the default `pdfRecord` record. For the Document Builder feature, this is a service report ID for a report that is in progress, queued, or failed. It must be used to generate a service document from failed state.

Outputs

Inputs	Details
`pdfReportId`	**Type** string **Description** Required. The report's record ID that holds the generated PDF. For service documents, the `pdfReportId` is a service report, and the record is created if the work order, work order line item, or service appointment is passed as the `recordId`.

Usage

Sample Input

The following code sample generates a PDF of a service document with a specific `recordId` and `templateId`:

```
{

  "inputs":[

    {

      "recordId": "08pOG00000023anYAA",
```

```
    "templateId": "0M0OG0000005Na40AE",
    "locale": "en_US",
    "title": "My Awesome PDF"
   }
  ]
}
```

Create Service Report Actions

Creates a service report for a service appointment, work order, or work order line item.

This object is available in API version 39.0 and later.

Supported REST HTTP Methods

URI
`/services/data/vXX.X/actions/standard/createServiceReport`

Formats
JSON

HTTP Methods
POST

Authentication
`Authorization: Bearer token`

Inputs

Input	Details
`entityId`	**Type** reference **Description** Required. The ID of the service appointment, work order, or work order line item that the service report is created for.
`signatures`	**Type** string **Description** Optional. A list of JSON definitions for a digital signature. • data—(Required) The base64 code for an image. • contentType—(Required)The file type of the signature.

Input	Details
	• signatureType—(Required) The role of the person signing; for example, "Customer." Salesforce admin defines Signature Type picklist values ahead of time. Each signature block must use a different signature type, and the signature types you define in your call must match the service report template's signature types. • name—The signature block title. This value appears on the generated service report. • place—The place of signing. This value appears on the generated service report. • signedBy—The name of the person signing. This value appears on the generated service report. • signedDate—The date of signing. This value appears on the generated service report.
`templateId`	**Type** reference **Description** Required. The ID of the standard or custom service report template that is used to create the service report.

Usage

Sample Input

The following code sample creates a service report with two signatures by making an Apex callout to the *createServiceReport* action REST API resource.

```
{
   "inputs" : [ {
      "entityId" : "0WOxx000000001E",
         "signatures" : [
                {"data":"Base64 code for the captured signature image",
                "contentType":"image/png",
                "name":"Customer Signature",
                "signatureType":"Customer",
                "place":"San Francisco",
                "signedBy":"John Doe",
                "signedDate":"Thu Jul 13 22:34:43 GMT 2017"
                },
                {"data":"Base64 code for the captured signature image",
                "contentType":"image/png",
                "name":"Technician Signature",
                "signatureType":"Technician"
                }],
         "templateId" : "0SLR00000004DBFOA2"
   } ]
}
```

Custom Notification Actions

Send custom notifications to recipients via desktop or mobile channels.

Before you send a custom notification, you must first create a notification type.

Important: In orgs created in Winter '21 or later, the Send Custom Notifications user permission is required to trigger the Send Custom Notification action in flows that run in user context, REST API calls, and Apex callouts.

The Send Custom Notifications user permission isn't required to trigger the Send Custom Notification action in processes or flows that run in system context.

This object is available in API version 46.0 and later.

Supported REST HTTP Methods

URI

`/services/data/vXX.X/actions/standard/customNotificationAction`

Formats

JSON, XML

HTTP Methods

GET, HEAD, POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`customNotifTypeId`	**Type** reference **Description** Required. The ID of the Custom Notification Type being used for the notification.
`recipientIds`	**Type** reference **Description** Required. The ID of the recipient or recipient type of the notification. Valid recipient or recipient type values are: • `UserId` — The notification is sent to this user, if this user is active. • `AccountId` — The notification is sent to all active users who are members of this account's Account Team. This recipient type is valid if account teams are enabled for your org. • `OpportunityId` — The notification is sent to all active users who are members of this opportunity's Opportunity Team.

Input	Details
	This recipient type is valid if team selling is enabled for your org. • `GroupId` — The notification is sent to all active users who are members of this group. • `QueueId` — The notification is sent to all active users who are members of this queue. Values can be combined in a list up to the maximum of 500 values.
`senderId`	**Type** reference **Description** Optional. The User ID of the sender of the notification.
`title`	**Type** string **Description** Required. The title of the notification, as seen by recipients. Maximum characters: 250. The content of mobile push notifications depends on the content push notification setting.
`body`	**Type** string **Description** Required. The body of the notification, as seen by recipients. Maximum characters: 750. The content of mobile push notifications depends on the content push notification setting..
`targetId`	**Type** reference **Description** Optional. The Record ID for the target record of the notification. You must specify either a `targetID` or a `targetPageRef`.
`targetPageRef`	**Type** string **Description** Optional. The `PageReference` for the navigation target of the notification. To see how to specify the target using JSON, see pageReference. You must specify either a `targetID` or a `targetPageRef`.

Usage

GET

The following example shows how to get information about the custom notification action type:

```
curl --include --request GET \
--header "Authorization: Authorization: Bearer 00DR...xyz" \
--header "Content-Type: application/json" \
"https://instance.salesforce.com/services/data/v46.0/actions/standard/customNotificationAction"
```

POST

The following example shows how to create a custom notification action:

```
curl --include --request POST \
--header "Authorization: Authorization: Bearer 00DR...xyz" \
--header "Content-Type: application/json" \
--data '{ "inputs" :
  [
  {
    "customNotifTypeId" : "0MLR0000000008eOAA",
    "recipientIds" : ["005R0000000LSqtIAG"],
    "title" : "Custom Notification",
    "body" : "This is a custom notification.",
    "targetId" : "001R0000003fSUDIA2"
  }
  ]
}' \
"https://instance.salesforce.com/services/data/v46.0/actions/standard/customNotificationAction"
```

The response is:

```
[
   {
      "actionName" : "customNotificationAction",
      "errors" : null,
      "isSuccess" : true,
      "outputValues" : {
         "SuccessMessage" : "Your custom notification is processed successfully."
      }
   }
]
```

Deploy Data Kit Components Action

Deploy data kit components sequentially in a subscriber org. The `deployDataKitComponents` invocable action is used in the `Deploy Data Kit Component` flow to invoke the deployment of each data kit component. This action is available for flows in API version 61.0 and later.

Inputs

Input	Details
dataKitComponentDeployInput	**Type** Apex-defined sfdatakit__DeployComponentInput

Input	Details
	Description Required. An Apex `sfdatakit__DeployComponentInput` payload that contains details about the components to deploy and their metadata.
`dataKitName`	**Type** text **Description** Required. The data kit name that contains the components to get the status for.
`dataSpaceName`	**Type** text **Description** The name of the data space to deploy the data kit to. If a data space isn't defined, the system deploys the components in the default data space.

Outputs

None.

Example

The following example shows a sample input payload for the different data kit components.

Input	Details
`sfdatakit__DeployComponentInput.bundleConfig`	**Type** Apex-defined **Input Payload**

```
{
    "inputs": [
        {
            "dataKitComponentsInput": [
                {
                    "componentType": "DataStreamBundle",
                    "bundleConfig": {
                        "connectorType": "CRM",
                        "bundleName": "hello", #full Qualified
 Bundle Name with namespace in datakit
                        "forceNoRefresh": true,
                        "bundleCRMConfig": {
                            "orgId": "org123" #Data Org Id
                        }
                    }
                }
            ],
```

Input	Details

```
            "dataKitNameInput": "datakit1",
            "dataKitDataSpaceInput" : "default"
        }
    ]
}
```

sfdatakit_DeployComponentBundleConnectorFrameworkConfig

Type

Apex-defined

Input Payload

```
{
    "inputs": [
        {
            "dataKitComponentsInput": [
                {
                    "componentType": "DataStreamBundle",
                    "bundleConfig": {
                        "connectorType": "MORECONNECTORS",
                        "bundleName": "hello", #full Qualified
 Bundle Name with namespace in datakit
                        "forceNoRefresh": true,
                        "bundleConnectorFrameworkConfig": {
                            "connectionName": "name"
                        }
                    }
                }
            ],
            "dataKitNameInput": "datakit1",
            "dataKitDataSpaceInput" : "default"
        }
    ]
}
```

sfdatakit_DeployComponentBundleIngestApiConfig

Type

Apex-defined

Input Payload

```
{
    "inputs": [
        {
            "dataKitComponentsInput": [
                {
                    "componentType": "DataStreamBundle",
                    "bundleConfig": {
                        "connectorType": "INGESTAPI",
                        "bundleName": "hello", #full Qualified
 Bundle Name with namespace in datakit
                        "forceNoRefresh": true,
                        "bundleIngestApiConfig": {
                            "connectorName": "name" #ingestion
```

Input	Details

```
 API connector name
                        }
                    }
                }
            ],
            "dataKitNameInput": "datakit1",
            "dataKitDataSpaceInput" : "default"
        }
    ]
}
```

sfdatakit__DeployComponentBundleConfig.bundleStreamingAppConfig

Type

Apex-defined

Input Payload

```
{
    "inputs": [
        {
            "dataKitComponentsInput": [
                {
                    "componentType": "DataStreamBundle",
                    "bundleConfig": {
                        "connectorType": "STREAMINGAPP",
                        "bundleName": "hello", #full Qualified
 Bundle Name with namespace in datakit
                        "forceNoRefresh": true,
                        "bundleStreamingAppConfig": {
                           "connectorName": "name", #Streaming
 app connector name
                            "streamingAppDataConnectorType":
"MobileApp" #MobileApp,WebApp
                        }
                    }
                }
            ],
            "dataKitNameInput": "datakit1",
            "dataKitDataSpaceInput" : "default"
        }
    ]
}
```

sfdatakit__DeployComponentInput.calculatedInsightsConfig

Type

Apex-defined

Input Payload

```
{
    "inputs": [
        {
            "dataKitComponentsInput": [
                {
```

Input	Details

```
                    "componentType": "CalculatedInsight",
                    "calculatedInsightsConfig": {
                       "apiName": "crm",#full Qualified CI Name
 with namespace in datakit
                        "apiNameOverride": "hello", #api name
of CI to be created on org
                        "label": "lab", #label of CI to be
created on org
                        "publishInterval":"NotScheduled"
#NotScheduled,One,Six,Twelve,TwentyFour
                    }
                }
            ],
            "dataKitNameInput": "datakit1",
            "dataKitDataSpaceInput" : "default"
        }
    ]
}
```

sfdatakit__DeployComponentInput.dloConfig

Type

Apex-defined

Input Payload

```
{
    "inputs": [
        {
            "dataKitComponentsInput": [
                {
                    "componentType": "DataLakeObject",
                    "dloConfig": {
                        "dataSourceObjectDevName": "crm",#full
 Qualified DLO Name with namespace in datakit
                        "apiName": "hello",#api name of DLO to
 be created on org
                      "label": "lab"#label of DLO to be created
 on org
                    }
                }
            ],
            "dataKitNameInput": "datakit1",
            "dataKitDataSpaceInput" : "default"
        }
    ]
}
```

sfdatakit__DeployComponentInput.dataTransformConfig

Type

Apex-defined

Input	Details
	Input Payload

```
{
    "inputs": [
        {
            "dataKitComponentsInput": [
                {
                    "componentType": "DataTransform",
                    "dataTransformConfig": {
                    "dataTransformType": "BATCH",
#STREAMING,BATCH
                    "dataTransformDevName":
"BatchTransformAccount",#full Qualified Transform Name with
namespace in datakit
                   "apiName":"BatchTransformAccount",#api name
 of Transform to be created on org
                    "label" : "BatchTransformAccount"#label of
 Transform to be created on org
                    }
                }
            ],
            "dataKitNameInput": "datakit1",
            "dataKitDataSpaceInput" : "default"
        }
    ]
}
```

Email Alert Actions

Send emails from flows by reusing already-configured workflow email alerts.

The email alert is already configured with the email's contents, recipients, and sender, so the flow only has to provide the record ID. Email alerts are entity-specific. For more information about creating email alerts, see Creating Email Alerts for Workflow, Approvals, or Milestones in Salesforce Help. Make sure to review the daily limits for emails sent from email alerts.

This object is available in API version 32.0 and later.

Supported REST HTTP Methods

URI

Get a list of available email alert actions:

`/services/data/vXX.X/actions/custom/emailAlert`

Get information about a specific email alert action:

`/services/data/vXX.X/actions/custom/emailAlert/entity_name/action_name`

Formats

JSON, XML

HTTP Methods
GET, HEAD, POST

Authentication
`Authorization: Bearer` ***`token`***

Inputs

The email alert action uses the record specified by `SObjectRowId` to get the information it needs. For example, if a Case was specified for the action, the action could retrieve the email address and recipient's name from the Case object's *`SuppliedEmail`* and *`SuppliedName`* fields, respectively.

Input	Details
`SObjectRowId`	**Type** ID **Description** Required. The ID of a record such as an Account.

Outputs

None.

Einstein Bots Actions

Search for knowledge articles based on data category and data category groups. To use these actions, you must enable Einstein Bots and Lightning Knowledge.

Get Data Category Details

Gets the labels and API names for a specified data category associated with the Knowledge object and its child categories.

This object is available in API version 56.0 and later.

Supported REST HTTP Methods

URI
`/services/data/56.0/actions/standard/getDataCategoryDetails`

Formats
JSON, XML

HTTP Methods
GET, HEAD, POST

Authentication
`Authorization: Bearer token`

Inputs

Input	Details
`dataCategoryGroupName`	**Type** string **Description** The API name of the data category group.
`dataCategoryName`	**Type** string **Description** The API name of the data category.

Outputs

Output	Details
`dataCategoryDetailsOutput`	**Type** Apex **Description** An Apex knowledge_bot__DataCategoryDetailsOutput record that contains the labels and API names for the data category and its child categories.

Get Data Category Groups

Gets the labels and API names of the active data category groups associated with the Knowledge object that are visible to the current user.

This object is available in API version 56.0 and later.

Supported REST HTTP Methods

URI

`/services/data/v56.0/actions/standard/getDataCategoryGroups`

Formats

JSON, XML

HTTP Methods

GET, HEAD, POST

Authentication

`Authorization: Bearer token`

Inputs

None

Outputs

Output	Details
`dataCategoryGroupInfo`	**Type** Apex **Description** An Apex knowledge_bot__DataCategoryGroupInfo record that contains the labels and API names of the data category groups visible to the current user.

Search Knowledge Articles

Searches for knowledge articles with specified search terms, language, data category group, and data category.

This object is available in API version 56.0 and later.

Supported REST HTTP Methods

URI

`/services/data/v56.0/actions/standard/searchKnowledgeArticles`

Formats

JSON, XML

HTTP Methods

GET, HEAD, POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`searchText`	**Type** string **Description** Search terms to be used in the knowledge article search.
`languageCode`	**Type** string **Description** The language code to be used in a knowledge article search. Valid values are language codes for fully supported languages.

Input	Details
`dataCategoryGroupName`	**Type** string **Description** The API name of the data category group to be used in the knowledge article search.
`dataCategoryName`	**Type** string **Description** The API name of the data category to be used in the knowledge article search.
`resultsLimit`	**Type** integer **Description** Optional. The maximum number of knowledge articles to return. Valid values are from 1 through 2000. By default, the maximum is 2000.

Outputs

Output	Details
`knoweldgeArticlesList`	**Type** Apex **Description** An Apex knowledge_bot__ArticlesList record that contains information about the knowledge articles that were returned.

Flow Actions

Invoke an active autolaunched flow that exists in the current org.

For more information about creating flows, see Build a Flow in Salesforce Help.

This object is available for autolaunched flows in API version 32.0 and later.

Supported REST HTTP Methods

URI

Get a list of available flow actions:

`/services/data/vXX.X/actions/custom/flow`

Invokes the `LargeOrder` flow:

`/services/data/vXX.X/actions/custom/flow/LargeOrder`

Formats

JSON, XML

HTTP Methods

GET, HEAD, POST

Authentication

`Authorization: Bearer token`

Inputs

Input values vary according to the input variables specified for each flow. For autolaunched flows, the input values vary according to the input variables in that flow.

Outputs

The Response to an invocation will include `Flow__InterviewStatus` and any output variables defined in the flow. For more information, see Flow Resource: Variable in Salesforce Help.

Output	Details
`Flow__InterviewStatus`	**Type** picklist **Description** The status of the flow interview. Valid values are: • Created • Started • Finished • Error • Waiting

Legacy Support for Process Builder

Processes created with type 'Invocable' in Process Builder can also be invoked via REST, using the endpoint listed above. See Create a Process in Salesforce Help. This object is available for invocable processes in API version 38.0 and later.

Invocable processes always require one of these input parameters:

- sObject: The sObject itself that you want the process to execute on. The sObject must be of the same object type as the one on which the process is defined.
- sObjectId: The Id of the sObject record that you want the process to execute on. The record must be of the same object type as the one on which the process is defined.

Invocable processes don't have outputs.

SEE ALSO:

Apex Developer Guide : InvocableMethod Annotation

REST API Developer Guide : Invocable Actions

Generate Order Summary Action

Generates a URL so that authenticated and guest users can access order details.

To access, you need these permissions.

- Salesforce Order Management License or Salesforce B2B Commerce License

This object is available in API version 59.0 and later.

Supported REST HTTP Methods

URI
`/services/data/v59.0/actions/standard/generateOrderSummaryUrl`

Formats
JSON

HTTP Methods
POST

Inputs

Input	Details
`orderSummaryId`	**Type** ID **Description** Optional. The ID of the order summary to generate a URL for.
`orderNumber`	**Type** string **Description** Optional. The ID of the Salesforce payment gateway record that represents the external payment gateway used for processing the sale request.
`webStoreId`	**Type** ID **Description** Required. The order number of the order summary to generate a URL for.

Outputs

Output	Details
`url`	**Type** string

Output	Details
	Description The URL generated by the action that links to the order summary.

Generate Work Orders Actions

Generates work orders from a maintenance plan. This object supports manual work order generation only. Before using this object, make sure that Auto-Generate Work Orders isn't selected on the maintenance plan.

This object is available in API version 40.0 and later.

Supported REST HTTP Methods

URI
`/services/data/vXX.X/actions/standard/generateWorkOrders`

Formats
JSON, XML

HTTP Methods
POST

Authentication
`Authorization: Bearer token`

Inputs

Input	Details
`recordId`	**Type** reference **Description** The ID of the maintenance plan from which you want to generate work orders.

Get Assessment Response Summary

Get Assessment Response Summary makes it easy to use a flow to trigger server-side document generation using Docgen.

In Discovery Framework, the responses from an assessment are stored in the AssessmentQuestionResponse object and the form metadata stays in OmniScript. You can use this invocable action to pass assessment summary data to the downstream processes. This invocable action provides a summary JSON that can be consumed in Docgen workflows to generate documents.

The Get Assessment Response Summary invocable action takes assessment ID as the input to get the OmniProcess ID, which is used to retrieve the OmniProcess elements. The assessment ID also retrieves the assessment response and merges the response with the OmniProcess elements to create an assessment summary response in JSON.

This object is available in API version 56.0 and later.

Supported REST HTTP Methods

URI
`/services/data/v56.0/actions/standard/getAssessmentResponseSummary`

Formats
JSON

HTTP Methods
POST

Authentication
`Authorization: Bearer token`

Inputs

Input	Details
`assessmentId`	**Type** ID **Description** Required. The ID of the assessment record for which to summarize responses.

Outputs

Output	Details
`assessmentResponseSummary`	**Type** string **Description** A JSON string containing the summary assessment question texts and responses for the specified assessment record. The response summary structure follows the structure of the OmniScript.

Usage

Sample Input

When exposing the Get Assessment Response Summary invocable action in a REST API, you can use the following format to pass input, which includes the assessmentId and its value.

```
{
  "inputs" : [ {"assessmentId" : "0U3RO00000005FN0AY"} ]
}
```

Sample Output

In this example, the first line indicates the OmniScript type, subtype, and language. For each step, there are multiple questions that appear in the OmniScript. You can use this information in a downstream process, such as generation of PDF document using Docgen capability.


```
  "KYC_Individual_English": {
    "Step1": {
      "label": "Identity Details",
      "value": {
        "LC_Survey_Question_2": {
          "label": "Full Name",
          "value": "Joe Smith"
        },
        "DateofBirth_m": {
          "label": "Date of Birth",
          "value": "Thu Jul 27 00:00:00 GMT 2000"
        },
        "Gender_m": {
          "label": "Gender",
          "value": "Female"
        },
        "EmailAddress_m": {
          "label": "Email Address",
          "value": "Joe.Smith@company.com"
        },
        "PAN": {
          "label": "PAN",
          "value": "QWEASDZXC"
        }
      }
    },
    "Step2": {
      "label": "Address Details",
      "value": {
        "Address_CorrespondenceAdd_Corporate": {
          "label": "Address of Correspondence",
          "value": "100 Some St, San Francisco, CA 12345, United States"
        },
        "Address_ContactDetails_Corporate": {
          "label": "Telephone/Mobile",
          "value": "1616111233"
        },
        "Alternate_Contact": {
          "label": "Alternate Mobile Number",
          "value": "1911212123"
        }
      }
    },
    "Step3": {
      "label": "Account Declaration",
      "value": {
        "Account_declaration": {
          "label": "I declare that I have following deposit accounts with your/
other bank's branches :",
```

```
          "value": [
            {
              "Bank": {
                "label": "Bank",
                "value": "Acme1"
              },
              "Branch": {
                "label": "Branch",
                "value": "Mission St"
              },
              "Type_of_Account": {
                "label": "Type of Account",
                "value": "Checking"
              },
              "Account_Number": {
                "label": "Account Number",
                "value": "12345678"
              }
            },
            {
              "Bank": {
                "label": "Bank",
                "value": "Acme2"
              },
              "Branch": {
                "label": "Branch",
                "value": "Mission St"
              },
              "Type_of_Account": {
                "label": "Type of Account",
                "value": "Savings"
              },
              "Account_Number": {
                "label": "Account Number",
                "value": "1234567890"
              }
            }
          ]
        }
      }
    },
    "Step4": {
      "label": "Declaration",
      "value": {
        "Declaration_m": {
          "label": "The customer declares and certifies that the information in this
form is true and correct. Any pre-filled sections of this form must be reviewed prior
 to signing and submitting, to ensure the information accurately conveys the new
account details.",
          "value": "true"
        }
      }
    }
```

```
    }
}
```

Knowledge Actions

Manage your Knowledge articles using invocable actions.

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.

The Assign and Publish actions are available in API version 44.0 and later. All the other actions are available in API version 45.0 and later.

Lightning Knowledge must be set up in your org. The user must have permissions to manage articles.

You can use multiple inputs to an invocable action. This technique is useful for actions that don't take lists, such as `restoreKnowledgeArticleVersion`.

Supported REST HTTP Methods

URIs

Archive Knowledge articles:

/services/data/v***XX.X***/actions/standard/archiveKnowledgeArticles

Assign Knowledge articles:

/services/data/v***XX.X***/actions/standard/assignKnowledgeArticles

Create draft from online Knowledge articles:

/services/data/v***XX.X***/actions/standard/createDraftFromOnlineKnowledgeArticle

Delete Knowledge articles:

/services/data/v***XX.X***/actions/standard/deleteKnowledgeArticles

Publish Knowledge articles:

/services/data/v***XX.X***/actions/standard/publishKnowledgeArticles

Restore Knowledge article version:

/services/data/v***XX.X***/actions/standard/restoreKnowledgeArticleVersion

Retrieve Smart Link URL:

/services/data/v***XX.X***/actions/standard/getArticleSmartLinkUrl

Submit Knowledge article for translation:

/services/data/v***XX.X***/actions/standard/submitKnowledgeArticleForTranslation

Formats

JSON, XML

HTTP Methods

GET, HEAD, POST

Authentication

`Authorization: Bearer` ***token***

Other Information

Error Response Types on page 53

Archive Knowledge Articles

URI: `/services/data/vXX.X/actions/standard/archiveKnowledgeArticles`

Table 1: Inputs

Input	Details
`articleVersionIdList`	**Type** string **Description** Required. Comma-separated article version ID list.

Sample Input

The following code sample archives two articles:

```
{
  "inputs" : [
    {
      "articleVersionIdList" : [ "ka0RM00000004VeYAI", "ka0RM00000003doYAA" ]
    }
  ]
}
```

Sample Output

The following code sample illustrates a response after a successful request.

```
[ {
  "actionName" : "archiveKnowledgeArticles",
  "errors" : null,
  "isSuccess" : true,
  "outputValues" : {
    "ka0RM00000004Ve" : "Success",
    "ka0RM00000003do" : "Success"
  }
} ]
```

The following code sample illustrates a response with one success and one failure:

```
[ {
  "actionName" : "archiveKnowledgeArticles",
  "errors" : null,
  "isSuccess" : false,
  "outputValues" : {
    "ka0RM00000004Ve" : "You can't perform this action. Be sure the action is valid for
 the current state of the article, and that you have permission to perform it.",
    "ka0RM00000003do" : "Success"
  }
} ]
```

Assign Knowledge Articles

URI: `/services/data/vXX.X/actions/standard/assignKnowledgeArticles`

Table 2: Inputs

Input	Details
`articleVersionIdList`	**Type** string **Description** Required. Comma-separated article version ID list.
`assigneeId`	**Type** ID **Description** Required. ID of the assigned user.
`assignAction`	**Type** string **Description** Required. Assign action. Valid actions are: • ASSIGN_DRAFT_MASTER • ASSIGN_DRAFT_TRANSLATION
`dueDate`	**Type** string **Description** Optional. Assigned due date.
`instruction`	**Type** string **Description** Optional. Instructions for the assignee.
`sendEmailNotification`	**Type** boolean **Description** Optional. Indicates whether to send an email notification. Defaults to `false`.

Sample Input

The following code sample assigns two articles for translation:

```
{
  "inputs" : [
```

```
    {
      "articleVersionIdList" : [ "ka0RM00000004VeYAI", "ka0RM00000003doYAA" ]
      "assigneeId" : "005RM00000AAAAAYA4",
      "assignAction" : "ASSIGN_DRAFT_TRANSLATION"
    }
  ]
}
```

Sample Output

The following code sample illustrates a response after a successful request.

```
[ {
  "actionName" : "assignKnowledgeArticles",
  "errors" : null,
  "isSuccess" : true,
  "outputValues" : {
    "ka0RM00000004Ve" : "Success",
    "ka0RM00000003do" : "Success"
  }
} ]
```

Create Draft from Online Knowledge Article

URI: `/services/data/v`***`XX.X`***`/actions/standard/createDraftFromOnlineKnowledgeArticle`

Table 3: Inputs

Input	Details
`action`	**Type** string **Description** Required. Edit action for primary language or translation articles. Valid actions are: • EDIT_AS_DRAFT_ARTICLE • EDIT_AS_DRAFT_TRANSLATION
`unpublish`	**Type** boolean **Description** Required. Indicates whether to keep the article published (`false`) or archive the published article (`true`). Use `false` to keep the current article version online and create a draft. Use `true` to archive the current online version, which removes it from the knowledge base, and creates a draft.
`articleVersionId`	**Type** string **Description** Article version ID. Required to create a draft from an online (published) translation. Optional to create a draft from the online primary article if the Article ID is provided.

Input	Details
articleId	**Type** string **Description** Article ID. Required when creating a draft from the online (published) primary article if the Article Version ID isn't provided.

Sample Input

The following code sample creates a draft from a primary article and archives the original article:

```
{
  "inputs" : [
    {
      "action" : "EDIT_AS_DRAFT_ARTICLE",
      "unpublish" : true,
      "articleId" : "kA0RM00000004pP0AQ"
    }
  ]
}
```

Sample Output

The following code sample illustrates a response after a successful request.

```
[ {
  "actionName" : "createDraftFromOnlineKnowledgeArticle",
  "errors" : null,
  "isSuccess" : true,
  "outputValues" : {
    "kA0RM00000004pP0AQ" : "Success"
  }
} ]
```

Delete Knowledge Articles

URI: `/services/data/vXX.X/actions/standard/deleteKnowledgeArticles`

Table 4: Inputs

Input	Details
articleVersionIdList	**Type** string **Description** Required. Comma-separated article version ID list.

Sample Input

The following code sample deletes two articles:

```
{
  "inputs" : [
    {
      "articleVersionIdList" : [ "ka0RM00000004VeYAI", "ka0RM00000003doYAA" ]
    }
  ]
}
```

Sample Output

The following code sample illustrates a response after a successful request.

```
[ {
  "actionName" : "deleteKnowledgeArticles",
  "errors" : null,
  "isSuccess" : true,
  "outputValues" : {
    "ka0RM00000004Ve" : "Success",
    "ka0RM00000003do" : "Success"
  }
} ]
```

Publish Knowledge Articles

URI: `/services/data/vXX.X/actions/standard/publishKnowledgeArticles`

Table 5: Inputs

Input	Details
`articleVersionIdList`	**Type** string **Description** Required. Comma-separated article version ID list.
`pubAction`	**Type** string **Description** Required. Publish action. Valid actions are: • PUBLISH_ARTICLE (which replaces the latest version) • PUBLISH_ARTICLE_NEW_VERSION (which creates a new version) • SCHEDULE_ARTICLE_FOR_PUBLICATION • PUBLISH_TRANSLATION
`pubDate`	**Type** string

Input	Details
	Description Optional. Scheduled publish date in ISO 8601 format `yyyy-MM-dd\'T\'HH:mm:ss.SSSZ`. For example, for February 8, 2023, 1:40 pm UTC+01:00 use `2023-02-08T13:40:00.000+0100`.

Sample Input

The following code sample publishes two articles:

```
{
  "inputs" : [
    {
      "articleVersionIdList" : [ "ka0RM00000004VeYAI", "ka0RM00000003doYAA" ],
      "pubAction" : "PUBLISH_ARTICLE"
    }
  ]
}
```

Sample Output

The following code sample illustrates a response after a successful request.

```
[ {
  "actionName" : "publishKnowledgeArticles",
  "errors" : null,
  "isSuccess" : true,
  "outputValues" : {
    "ka0RM00000004Ve" : "Success",
    "ka0RM00000003do" : "Success"
  }
} ]
```

Restore Knowledge Article Version

URI: `/services/data/vXX.X/actions/standard/restoreKnowledgeArticleVersion`

Table 6: Inputs

Input	Details
`action`	**Type** string **Description** Required. The only valid action is: RESTORE_KNOWLEDGE_ARTICLE_VERSION
`articleId`	**Type** string **Description** Required. Article ID.

Input	Details
`versionNumber`	**Type** integer **Description** Optional. Version number of the archived article version to restore. Default is the latest archived version.

Sample Input

The following code restores the latest archived version:

```
{
  "inputs" : [
    {
      "action" : "RESTORE_KNOWLEDGE_ARTICLE_VERSION",
      "articleId" : "kA0RM00000004pP0AQ"
    }
  ]
}
```

The following code restores a past archived version of a published article:

```
{
  "inputs" : [
    {
      "action" : "RESTORE_KNOWLEDGE_ARTICLE_VERSION",
      "versionNumber":3,
      "articleId" : "kA0RM00000004pP0AQ"
    }
  ]
}
```

The following code restores two archived articles:

```
{
  "inputs" : [
    {
      "action" : "RESTORE_KNOWLEDGE_ARTICLE_VERSION",
      "articleId" : "kA0RM00000004pP0AQ"
    },
    {
      "action" : "RESTORE_KNOWLEDGE_ARTICLE_VERSION",
      "articleId" : "kA0RM00000004pP0AB"
    }
  ]
}
```

Sample Output

The following code sample illustrates a response after a successful request.

```
[ {
  "actionName" : "restoreKnowledgeArticleVersion",
```

```
  "errors" : null,
  "isSuccess" : true,
  "outputValues" : {
    "kA0RM00000004pP0AQ" : "Success"
  }
} ]
```

Retrieve Smart Link URL

URI: `/services/data/vXX.X/actions/standard/getArticleSmartLinkUrl`

Table 7: Inputs

Input	Details
`articleVersionId`	**Type** string **Description** Required. The ID of the Knowledge article version.

Sample Input

The following code sample retrieves the SmartLink URL of a Knowledge article version:

```
{
   "inputs":[
      {
         "articleVersionId":"ka0xx00000000cjAAA"
      }
   ]
}
```

Sample Output

The following code sample illustrates a response after a successful request.

```
[
   {
      "actionName":"getArticleSmartLinkUrl",
      "errors":null,
      "isSuccess":true,
      "outputValues":{

"articleSmartLinkUrl":"https://example.lightning.force.com/lightning/articles/Knowledge/Test-Redirection-1"

      }
   }
]
```

Submit Knowledge Article for Translation

URI: `/services/data/vXX.X/actions/standard/submitKnowledgeArticleForTranslation`

Table 8: Inputs

Input	Details
`articleId`	**Type** string **Description** Required. Article ID.
`language`	**Type** string **Description** Required. Language code for the translation.
`assigneeId`	**Type** ID **Description** Required. ID of the assigned user.
`dueDate`	**Type** string **Description** Optional. Assigned due date.
`sendEmailNotification`	**Type** boolean **Description** Optional. Indicates whether to send an email notification. Defaults to `false`.

Table 9: Outputs

Output	Details
`articleId`	**Type** ID **Description** Article ID.
`language`	**Type** string **Description** Language code for the translation.

Sample Input

The following code sample submits one article for translation into Spanish:

```
{
  "inputs" : [
    {
      "articleId" : "kA0RM00000004pP0AQ",
      "language" : "es",
      "assigneeId" : "005RM00000AAAAAYA4"
    }
  ]
}
```

Sample Output

The following code sample illustrates a response after a successful request.

```
[ {
  "actionName" : "submitKnowledgeArticleForTranslation",
  "errors" : null,
  "isSuccess" : true,
  "outputValues" : {
    "articleId" : "kA0RM00000004pP0AQ",
    "language" : "es"
  }
} ]
```

Error Response Types

Knowledge actions can respond with two types of error responses: action-scoped errors and item-scoped errors.

Action-scoped errors describe an error about the overall action that you're trying to invoke. Action-scoped errors have a `statusCode` in addition to a `message`. This example illustrates an action-scoped error caused by sending invalid input values.

```
[ {
        "actionName" : "restoreKnowledgeArticleVersion",
        "errors" : [ {
        "statusCode" : "INVALID_API_INPUT",
        "message" : "You can't perform this action. Be sure the action is valid for the
current state of the article, and that you have permission to perform it.",
        "fields" : [ ]
        } ],
        "isSuccess" : false,
        "outputValues" : null
        } ]
```

Item-scoped errors describe a problem with a specific article or article version within the action. For example, this code illustrates an archiveKnowledgeArticles action response with one failed item and one successful item.

```
[ {
        "actionName" : "archiveKnowledgeArticles",
        "errors" : null,
        "isSuccess" : false,
        "outputValues" : {
        "ka0RM00000004Ve" : "You can't perform this action. Be sure the action is valid
```

```
for the current state of the article, and that you have permission to perform it.",
          "ka0RM00000003do" : "Success"
          }
          } ]
```

If any type of error occurs with an action, the `isSuccess` field is `false`.

Lead Action

Manage your leads using the invocable action.

Leads must be enabled in your org. The user must have read and edit permissions for leads.

Supported REST HTTP Methods

URIs

Apply Lead Assignment Rules:

`/services/data/vXX.X/actions/standard/invocableApplyLeadAssignmentRules`

Formats

JSON, XML

HTTP Methods

GET, HEAD, POST

Authentication

`Authorization: Bearer token`

Other Information

Error Response Types on page 55

Apply Lead Assignment Rules

Run lead assignment rules on a collection of leads.

Available in API version 54.0.

URI: `/services/data/vXX.X/actions/standard/invocableApplyLeadAssignmentRules`

Table 10: Inputs

Input	Details
`LeadIds`	**Type** String Collection **Description** Required. A collection of lead IDs to run lead assignment rules for.

Sample Input

The following code runs lead assignment rules for two leads:

```
{"inputs": [ {
      "leadId" : "00QR00000006LE8OAM"
    },
    {
        "leadId" : "00QR00000006LEDOA2"
    }]
}
```

Sample Output

The following code sample illustrates a response when the action succeeds.

```
[ {
          "actionName" : "invocableApplyLeadAssignmentRules",
          "isSuccess" : true
          } ]
```

Error Response Types

Sales Engagement actions can respond with success or errors.

If any type of error occurs with an action, the `isSuccess` field is `false`.

This example illustrates an error caused when the user has insufficient access to leads when calling the Apply Lead Assignment Rules action.

```
[ {
  "actionName" : "invocableApplyLeadAssignmentRules",
  "errors" : [ {
    "statusCode" : "INSUFFICIENT_ACCESS_OR_READONLY",
    "message" : "Looks like you don't have access to this record. Your Salesforce admin
can help with that.",
    "fields" : [ ]
  } ],
  "isSuccess" : false,
  "outputValues" : null
} ]
```

This example illustrates an error caused when the lead IDs passed to the Apply Lead Assignment Rules action are invalid.

```
[ {
  "actionName" : "invocableApplyLeadAssignmentRules",
  "errors" : [ {
    "statusCode" : "UNKNOWN_EXCEPTION",
    "message" : "Something's not right with one or more the specified LeadIds. Check the
IDs and try again.",
    "fields" : [ ]
  } ],
  "isSuccess" : false,
  "outputValues" : null
} ]
```

This example illustrates an error caused when one of the leads passed to the Apply Lead Assignment Rules action has been deleted.


```
[ {
  "actionName" : "invocableApplyLeadAssignmentRules",
  "errors" : [ {
    "statusCode" : "ENTITY_IS_DELETED",
    "message" : "One or more of the specified LeadIds were deleted. Check the IDs and try
 again.",
    "fields" : [ ]
  } ],
  "isSuccess" : false,
  "outputValues" : null
} ]
```


Live Message Notification Actions

Use messaging templates to send notifications to users over communication channels, such as SMS, WhatsApp, and Facebook Messenger, when certain conditions are met.

This action is available in API version 43.0 and later.

For more information about using Live Message Notification actions in flows, see Create Flows to Send Automatic Message Notifications in Salesforce Help.

Special Access Rules

To access Live Message Notification action for Surveys, you must have the Feedback Management Starter or Growth license and Salesforce org enabled with a default community.

Supported REST HTTP Methods

URI

`/services/data/v`***`43.0`***`/actions/standard/liveMessageNotification`

Formats

JSON, XML

HTTP Methods

GET, HEAD, POST

Authentication

`Authorization: Bearer` ***`token`***

Inputs

Input	Details
`channelDeveloperName`	**Type** string

Input	Details
	Description Required. The unique name of the messaging channel that's used to send the messaging notification.
`contextRecordId`	**Type** ID **Description** The entity ID of the sObject that's used as the context for the merge field in a messaging template. To resolve a merge field, the `contextRecordId` input property is required.
`conversationBroadcastId`	**Type** ID **Description** The entity ID of the sObject that links all the messages within a broadcast.
`recipientPhone`	**Type** string **Description** The destination phone number that the message is sent to. If the messaging channel type is SMS and the `recipientRecordId` input property isn't a messaging user, then the `recipientPhone` input property is required.
`recipientRecordId`	**Type** ID **Description** Required. The `MessagingEndUserId` property or the Record ID associated with a recipient phone number, to send the message notification.
`surveyDeveloperName`	**Type** string **Description** The API name of the survey that's sent in the message. Available in API version 57.0 and later. This input property is applicable to Surveys only.
`templateDeveloperName`	**Type** string **Description** Required. The unique name of the messaging template that's used to generate the message.
`triggeredOutboundTypeEnum`	**Type** picklist

Input	Details
	Description The type of triggered outbound message. Possible values are: • `Standard` • `Broadcast`

Outputs

None.

Example

GET

The following example shows how to get information about the Live Message Notification action type:

```
curl --include --request GET \
--header "Authorization: Authorization: Bearer 00DR...xyz" \
--header "Content-Type: application/json" \
"https://instance.salesforce.com/services/data/v43.0/actions/standard/liveMessageNotification"
```

POST

Here's an example request for the Live Message Notification action:

```
{
   "inputs":[{
      "templateDeveloperName":"MessageQ3Template",
      "channelDeveloperName":"MessageQ3Template",
      "contextRecordId":"003RO00000480RvYAM",
      "recipientRecordId":"0PARM000000GJzo4AG"
   }]
}
```

Here's an example response for the Live Message Notification action:

```
[
   {
      "actionName" : "liveMessageNotification",
      "errors" : null,
      "isSuccess" : true,
      "outputValues" : null
   }
]
```

Omni-Channel Action

Create a `PendingServiceRouting` record used for Omni-Channel skills-based routing.

For more information about skills-based routing, see Skills-Based Routing for Omni-Channel in Salesforce Help.

This object is available in API version 44.0 and later.

Supported REST HTTP Methods

URI

`/services/data/vXX.X/actions/standard/skillsBasedRouting`

Formats

JSON, XML

HTTP Methods

POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`recordId`	**Type** ID **Description** Required. ID of the Salesforce record to be routed by Omni-Channel.
`routingConfigId`	**Type** ID **Description** Required. ID of the `QueueRoutingConfig` record to be used by Omni-Channel.
`skillIdList`	**Type** string **Description** Optional. Comma-separated list of `Skill` IDs. Maximum number of skills is 25.

Outputs

Output	Details
`pendingServiceRoutingId`	**Type** ID **Description** ID of the new `PendingServiceRouting` record, if the request was successful.

Usage

Sample Input

The following code sample attempts to create a `PendingServiceRouting` record using a work record (`recordId`), the routing configuration (`routingConfigId`), and two skills (`skillIdList`).

```
{
  "inputs": [{
      "recordId":"400B0000004GGUUIA4",
      "routingConfigId":"0K8B0000000CbgZKAS",
      "skillIdList":"0C4B00000008QImKAM,0C4B0000000CcR1KAK"
  }]
}
```

Sample Output

The following code sample illustrates a response after a successful request.

```
[
  {
    "actionName":"skillsBasedRouting",
    "errors":null,
    "isSuccess":true,
    "outputValues":{
      "pendingServiceRoutingId":"0JRB0000000TWA2"
    }
  }
]
```

Apply Payment Action

Applies a payment record to an invoice header by creating a PaymentLineInvoice record with a type of Applied.

To access Commerce Payments resources, you need the following permissions.

- Salesforce Order Management License or Salesforce B2B Commerce License
- PaymentsAPIUser user permission. This permission is available with the Salesforce Order Management or B2B Commerce License. Your Salesforce admin assigns it to your user profile.

This object is available in API version 54.0 and later.

Supported REST HTTP Methods

URI

`/services/data/v54.0/actions/standard/applyPayment`

Formats

JSON

HTTP Methods

POST

Authentication

`Authorization: Bearer` ***`token`***

Inputs

Input	Details
`amount`	**Type** number **Description** Required. The amount to be applied to the invoice header.
`appliedToId`	**Type** ID **Description** Required. The ID of the invoice that receives the payment.
`associatedAccountId`	**Type** ID **Description** Optional. The ID of the account that contains the invoice.
`comments`	**Type** String **Description** Optional comments for more information about the payment application.
`effectiveDate`	**Type** datetime **Description** Optional. The date that the payment takes effect on the invoice.
`paymentId`	**Type** string **Description** Required. The payment that's applied to the invoice. The application is represented by the PaymentLineInvoice created for a successful action.

Outputs

Output	Details
`appliedDate`	**Type** datetime **Description** The date that the payment was applied to the invoice header.

Output	Details
paymentLineInvoiceId	**Type** ID **Description** Represents the application of the payment's amount to the invoice. Created after a successful action.

Payment Sale Action

Capture a payment without any prior authorization and create a payment record. The payment sale transaction consists of an authorize request and a capture request made to the payment gateway at the same time. This way, the merchant can request funds to be transferred to the merchant account in a single command, with no further action required.

To access Commerce Payments resources, you need the following permissions.

- Salesforce Order Management License or Salesforce B2B Commerce License
- PaymentsAPIUser user permission. This permission is available with the Salesforce Order Management or B2B Commerce License. Your Salesforce admin assigns it to your user profile.

The payment sale API handles only one payment at a time. Bulk requests aren't supported.

This object is available in API version 54.0 and later.

Supported REST HTTP Methods

URI
`/services/data/v54.0/actions/standard/paymentSale`

Formats
JSON

HTTP Methods
POST

Authentication
`Authorization: Bearer` ***token***

Inputs

Input	Details
amount	**Type** number **Description** Required. The amount of the payment sale request.
paymentGatewayId	**Type** ID

Input	Details
	Description Required. The ID of the Salesforce payment gateway record that represents the external payment gateway used for processing the sale request.
`paymentMethodId`	**Type** ID **Description** Required. The ID of the Salesforce payment method that contains customer payment information.
`currencyIsoCode`	**Type** string **Description** Required for multicurrency orgs. Three-letter ISO 4217 currency code associated with the payment output.
`idempotencyKey`	**Type** string **Description** Optional. Key used to ensure idempotency and avoid duplicate payments.

Outputs

Output	Details
`actionName`	**Type** string **Description** The name of the action performed. Becomes `paymentSale` following a Payment Sale action.
`errors`	**Type** string **Description** Following a `400` error response, the error objects show information about the error that occurred. Contains a status code, message, and list of fields.
`isSuccess`	**Type** boolean **Description** Shows whether the payment sale request was successful.

Output	Details
`outputValues`	**Type** ID **Description** The ID of the new payment request record.

Perform Survey Sentiment Analysis

Create or update the AI Sentiment Result records. You can get insights into the sentiments underlying survey responses and save the sentiment analysis in the SentimentAnalysisResult object.

This action is available in API version 55.0 and later.

Special Access Rules

To access the Perform Survey Sentiment Analysis action, you must have the Feedback Management - Starter and Feedback Management - Growth licenses.

Supported REST HTTP Methods

URI
`/services/data/v55.0/actions/standard/performSurveySentimentAnalysis`

Formats
JSON

HTTP Methods
POST

Authentication
`Authorization: Bearer token`

Inputs

Input	Details
`surveyId`	**Type** ID **Description** Required. The ID of the survey containing the questions for whose responses you want to get sentiment insights.
`surveyQuestionIds`	**Type** ID **Description** Required. The IDs of the questions for whose responses you want to get sentiment insights.

Input	Details
startDate	**Type** Datetime **Description** Required. The date from when participant responses are processed to get sentiment insights.
endDate	**Type** Datetime **Description** Required. The date until when participant responses are processed to get sentiment insights.
typeOfOperation	**Type** String **Description** Required. The type of operation to be performed on the survey responses. Possible values are: • create—Bulk process survey responses. After the processing is completed, the AI Sentiment Result records are created with the Submitted status. • update—Bulk process survey responses that have associated AI Sentiment Result records in the Draft status. After the processing is completed, the AI Sentiment Result records are updated and their status is changed to Submitted. You can only update a sentiment analysis result record with the Draft status.

Outputs

None.

Example

Sample Request

Here's an example POST request to create or update the AI Sentiment Result records:

```
{
   "inputs":[{
      "surveyId":"0Kdx00000000GYeCAM",
      "surveyQuestionIds":["0Kux00000000xDgCAI","0Kux00000000xDiCAI"],
      "startDate":"1-07-2022",
      "endDate":"12-07-2022",
      "typeOfOperation":"create"
```

```
    }]
}
```

SEE ALSO:

Salesforce Help: Automate Your Business Process: Perform Survey Sentiment Analysis

PlatformAction

PlatformAction is a virtual read-only object. It enables you to query for actions displayed in the UI, given a user, a context, device format, and a record ID. Examples include standard and custom buttons, quick actions, and productivity actions.

Supported Calls

`describeSObjects()`, `query()`

Fields

Field	Details
`ActionListContext`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** Required. The list context this action applies to. Valid values are: • `Assistant` • `BannerPhoto` • `Chatter` • `Dockable` • `FeedElement` • `Flexipage` • `Global` • `ListView` • `ListViewDefinition` • `ListViewRecord` • `Lookup` • `MruList` • `MruRow` • `ObjectHomeChart` • `Photo` • `Record`

Field	Details
	• `RecordEdit` • `RelatedList` • `RelatedListRecord`
`ActionTarget`	**Type** textarea **Properties** Nillable **Description** The URL to invoke or describe the action when the action is invoked. If the action is a standard button overridden by a Visualforce page, the ActionTarget returns the URL of the Visualforce page, such as `/apex/`*`pagename`*. This field is available in API version 35.0 and later.
`ActionTargetType`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** The type of the target when this action is triggered. Valid values are: • `Describe`—applies to actions with a user interface, such as quick actions • `Invoke`—applies to actions with no user interface, such as action links or invocable actions • `Visualforce`—applies to standard buttons overridden by a Visualforce page
`ActionTargetUrl`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** URL to invoke or describe the action when the action is invoked. This field is deprecated in API version 35.0 and later. Use `ActionTarget` instead.
`Category`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** Applies only to action links. Denotes whether the action link shows up in the feed item list of actions or the overflow list of actions. Valid values are: • `Primary`

Field	Details
	• `Overflow`
`ConfirmationMessage`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** Applies only to action links. The message to display before the action is invoked. Field is null if no confirmation is required before invoking the action.
`DeviceFormat`	**Type** picklist **Properties** Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort **Description** Specifies which action icon the PlatformAction query returns. If this field isn't specified, it defaults to Phone. Valid values are: • `Aloha` • `Desktop` • `Phone` • `Tablet`
`ExternalId`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The unique ID for the PlatformAction. If the action doesn't have an ID, its API name is used.
`GroupId`	**Type** reference **Properties** Filter, Group, Nillable, Sort **Description** The unique ID of a group of action links.
`IconContentType`	**Type** string **Properties** Filter, Group, Nillable, Sort

Field	Details
	Description The content type—such as .jpg, .gif, or .png—of the icon for this action. Applies to both custom and standard icons assigned to actions.
`IconHeight`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The height of the icon for this action. Applies only to standard icons.
`IconUrl`	**Type** url **Properties** Filter, Group, Nillable, Sort **Description** The URL of the icon for this action.
`IconWidth`	**Type** int **Properties** Filter, Group, Nillable, Sort **Description** The width of the icon for this action. Applies only to standard icons.
`InvocationStatus`	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** The status of the action within the feed item. Applies to action links only. Valid values are: • `Failed` • `New` • `Pending` • `Successful`
`InvokedByUserId`	**Type** reference **Properties** Filter, Group, Nillable, Sort

Field	Details
	Description The ID of the user who most recently invoked this action within the current feed item. Applies to action links only. This is a relationship field. **Relationship Name** InvokedByUser **Relationship Type** Lookup **Refers To** User
`IsGroupDefault`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Denotes whether this action is the default in an action link group. False for other action types. Applies to action links only.
`IsMassAction`	**Type** boolean **Properties** Defaulted on create, Filter, Group, Sort **Description** Indicates whether the action can be performed on multiple records. This field is available in API version 38.0 and later.
`Label`	**Type** string **Properties** Filter, Group, Sort **Description** The label to display for this action.
`PrimaryColor`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The primary color of the icon for this action.

Field	Details
RelatedListRecordId	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** Represents the ID of a record in an object's related list. This field is available in API version 38.0 and later.
RelatedSourceEntity	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** When the `ActionListContext` is RelatedList or RelatedListRecord, this field represents the API name of the related list to which the action belongs.
Section	**Type** picklist **Properties** Filter, Group, Nillable, Restricted picklist, Sort **Description** The section of the user interface the action resides in. Applicable only to Lightning Experience. Valid values are: • ActivityComposer • CollaborateComposer • NotesComposer • Page • SingleActionLinks This field is available in API version 35.0 and later.
SourceEntity	**Type** string **Properties** Filter, Group, Sort **Description** Required. The object or record with which this action is associated.
Subtype	**Type** string **Properties** Filter, Group, Nillable, Sort

Field	Details
	Description The subtype of the action. For quick actions, the subtype is `QuickActionType`. For custom buttons, the subtype is `WebLinkTypeEnum`. For action links, subtypes are `Api`, `ApiAsync`, `Download`, and `Ui`. Standard buttons and productivity actions have no subtype.
`TargetObject`	**Type** string **Properties** Filter, Group, Nillable, Sort **Description** The type of object record the action creates, such as a contact or opportunity. This field is available in API version 41.0 and later.
`TargetUrl`	**Type** string **Properties** Filter, Nillable, Sort **Description** The URL that a custom button or link points to. This field is available in API version 41.0 and later.
`Type`	**Type** picklist **Properties** Filter, Group, Restricted picklist, Sort **Description** The type of the action. Valid values are: • `ActionLink`—An indicator on a feed element that targets an API, a web page, or a file, represented by a button in the Salesforce Chatter feed UI. • `CustomButton`—When clicked, opens a URL or a Visualforce page in a window or executes JavaScript. • `InvocableAction` • `ProductivityAction`—Productivity actions are predefined and attached to a limited set of objects. Productivity actions include Send Email, Call, Map, View Website, and Read News. Except for the Call action, you can’t edit productivity actions. • `QuickAction`—A global or object-specific action. • `StandardButton`—A predefined Salesforce button such as New, Edit, and Delete.

Usage

PlatformAction can be described using describeSObject().

You can directly query for PlatformAction. For example, this query returns all fields for actions associated with each of the records of the listed objects:

```
SELECT ExternalId, ActionTargetType, ActionTargetUrl, ApiName, Category,
      ConfirmationMessage, ExternalId, GroupId, UiTheme, IconUrl, IconContentType,
      IconHeight, IconWidth, PrimaryColor, InvocationStatus, InvokedByUserId,
      IsGroupDefault, Label, LastModifiedDate, Subtype, SourceEntity, Type
FROM PlatformAction
WHERE SourceEntity IN ('001xx000003DGsH', '001xx000003DHBq', 'Task') AND
      ActionListContext = 'Record';
```

Note: To query PlatformAction, provide the `ActionListContext` and `SourceEntity`. If you query for `ActionListContext` with a value of `RelatedList`, and don't specify a `RelatedSourceEntity`, the query returns the API name of the related list. In API v43.0 and before, `SourceEntity = '`***`Object API Name`***`' and ActionListContext = 'ListView'` is an invalid combination to fetch quick actions in a SOQL query. Use `SourceEntity = '`***`Object ID`***`' and ActionListContext = 'ListView'` instead.

This query uses multiple `ActionListContext` values in its `WHERE` clause to return all actions in the Lightning Experience user interface (`DeviceFormat = 'Desktop'`) for the specified object:

```
SELECT ActionListContext, Label, Type, Subtype, Section, SourceEntity,
     RelatedSourceEntity, ActionTarget, ActionTargetType, ApiName, Category,
     ConfirmationMessage, DeviceFormat, ExternalId, GroupId, IconContentType,
     IconHeight, IconUrl, IconWidth, Id, InvocationStatus, InvokedByUserId,
     IsGroupDefault, LastModifiedDate, PrimaryColor
FROM PlatformAction
WHERE ActionListContext IN ('Record','Chatter','RelatedList') AND
      SourceEntity = '001xx000003DlvX' AND
      DeviceFormat = 'Desktop'
```

Post to Chatter Actions

Post to the feed for a specific record, user, or Chatter group.

Use a Post to Chatter action to post a message at run time to a specified feed. Post to Chatter supports @mentions and topics, but only text posts are supported.

This object is available in API version 32.0 and later.

Supported REST HTTP Methods

URI

Get a list of available post to Chatter actions:

`/services/data/v`***`XX.X`***`/actions/standard/chatterPost`

Formats

JSON, XML

HTTP Methods

GET, HEAD, POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`communityId`	**Type** reference **Description** Optional. Specifies the ID of an Experience Cloud site to post to. Valid only if Digital Experiences is enabled. Required if posting to a user or Chatter group that belongs to an Experience Cloud site. This value is available in API version 35.0 and later.
`subjectNameOrId`	**Type** string **Description** Required. Reference to the user, Chatter group, or record whose feed you want to post to. • To post to a user's feed, enter the user's `ID` or `Username`. For example: *`jsmith@salesforce.com`* • To post to a Chatter group, enter the group's `Name` or `ID`. For example: *`Entire Organization`* • To post to a record, enter the record's `ID`. For example: *`001D000000JWBDx`*
`text`	**Type** string **Description** Required. The text that you want to post. Must be a string of no more than 10,000 characters. To mention a user or group, enter @[`reference`], where `reference` is the ID for the user or group that you want to mention. The reference can be a literal value, a merge field, or a flow resource. To add a topic, enter #[`string`], where `string` is the topic that you want to add. For example, the string `Hi @[005000000000001] check this out #[some_topic].` is stored appropriately as `Hi @Joe, check this out #some_topic.` where "@Joe" and "#some_topic" are links to the user and topic, respectively.
`type`	**Type** picklist **Description** Required only if `subjectNameOrId` is set to a username or a Chatter group name. The type of feed that you want to post to. • *`User`*—Enter this value if `subjectNameOrId` is set to a user's `Username`. • *`Group`*—Enter this value if `subjectNameOrId` is set to a Chatter group's `Name`.

Input	Details
visibility	**Type** picklist **Description** Optional. Valid only if Digital Experiences is enabled. Specifies whether this feed item is available to all users or internal users only. Valid values are: • *allUsers* • *internalUsers* This value is available in API version 35.0 and later.

Outputs

Output	Details
feedItemId	**Type** reference **Description** The ID of the new Chatter feed item.

Preview Cart to Exchange Order

Generate preview details of an exchange order for specified order summary, exchange cart ID, and reference record ID.

To access, you need the following permissions.

- Salesforce Order Management License or Salesforce B2B Commerce License

This object is available in API version 60.0 and later.

Supported REST HTTP Methods

URI
`/services/data/v59.0/actions/standard/previewCartToExchangeOrder`

Formats
JSON, XML

HTTP Methods
GET, HEAD, POST

Authentication
Authorization: Bearer token

Inputs

Input	Details
`orderSummaryId`	**Type** ID **Description** Required. The ID of the order summary record associated with the list of exchanges.
`exchangeCartId`	**Type** ID **Description** Required. The ID of the cart record containing the items for the exchange order.
`referenceId`	**Type** ID **Description** Required. The ID of the record that's related to the specified order summary. Only IDs from a return order record are supported.

Outputs

Output	Details
`changeBalances`	**Type** string **Description** A string that contains the calculated amounts resulting from the exchange order.
`errors`	**Type** string **Description** Following a `400` error response, the error objects show information about the error that occurred. Contains a status code, message, and list of fields.
`isSuccess`	**Type** boolean **Description** Shows whether the payment sale request was successful.
`orderSummaryID`	**Type** ID

Output	Details
	Description The ID of the order summary record associated with the list of exchanges.

Prompt Template Actions

Creates a response based on the large language model (LLM) response for the specified prompt template and inputs.

This object is available in API version 60.0 and later.

This action is available only if you enable Prompt Builder and the user who runs the flow has the Prompt Template User permission.

The API name for each action is prefixed with `generatePromptResponse`.

Supported REST HTTP Methods

URI

Get a list of available Prompt Template actions:
`/services/data/vXX.X/actions/custom/generatePromptResponse`

Get information about a specific Prompt Template action:
`/services/data/vXX.X/actions/custom/generatePromptResponse/`***`template_API_name`***

Formats

JSON, XML

HTTP Methods

GET, HEAD, POST

Authentication

`Authorization: Bearer` ***`token`***

Inputs

Input	Details
`citation`	**Type** `Picklist` **Description** Specifies how citations are returned: • `post_generation`: Return citations after generation • `off`: Don't return citations
`promptResponse`	**Type** string **Description** The prompt response generated by the action based on the specified prompt template and input.

Additional input values vary according to the input variables specified for the prompt template.

Outputs

Output	Details
`citation`	**Type** `AiCopilot.GenAiCitationOutput` **Description** The prompt response generated by the action based on the specified prompt template and input.
`promptResponse`	**Type** string **Description** The prompt response generated by the action based on the specified prompt template and input.

Quick Actions

Use a quick action to create a task or a case. Invoke existing quick actions, both global and object-specific, to create records, update records, or log calls.

For more information about creating global quick actions, see Create Global Quick Actions, and for more information on object-specific quick actions, see Create Object-Specific Quick Actions, in Salesforce Help.

This object is available in API version 32.0 and later.

Supported REST HTTP Methods

URI

Get a list of quick actions:

`/services/data/vXX.X/actions/custom/quickAction`

Get a specific quick action:

`/services/data/vXX.X/actions/custom/quickAction/quick_action_name`

Formats

JSON, XML

HTTP Methods

GET, HEAD, POST

Authentication

`Authorization: Bearer token`

Notes

The only type of quick actions that can be invoked are create, update, and logACall.

Inputs

All quick actions have the `contextId` input parameter. It's a reference to the related record for the quick action. Other inputs vary according to the layout of the quick action. To determine inputs for a specific quick action, use the describe feature. For example, perform a GET with `/services/data/vXX.X/actions/custom/quickAction/Task/deferTask` to see the inputs for the quick action `deferTask`.

Refresh Metric Actions

Update a metric's Current Value field if it's linked to a summary field in a Salesforce report. The refresh runs as the metric owner.

This object is available in API version 34.0 and later.

Supported REST HTTP Methods

URI

Get a list of metric refresh actions:

`/services/data/vXX.X/actions/standard/metricRefresh`

Formats

JSON, XML

HTTP Methods

GET, HEAD, POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`metricId`	**Type** string **Description** Required. The metric linked to a Salesforce report.

Outputs

Output	Details
`resultingMessage`	**Type** string **Description** The message that indicates the results of the metric refresh.

Sales Engagement Actions

Manage your Sales Engagement cadences using invocable actions.

Sales Engagement must be set up in your org. The user must have permissions to use cadences.

Supported REST HTTP Methods

URIs

Assign Target To Cadence:

`/services/data/vXX.X/actions/standard/assignTargetToSalesCadence`

Remove Target From Cadence:

`/services/data/vXX.X/actions/standard/removeTargetFromSalesCadence`

Pause Cadence Tracker:

`/services/data/vXX.X/actions/standard/pauseSalesCadenceTracker`

Resume Cadence Tracker:

`/services/data/vXX.X/actions/standard/resumeSalesCadenceTracker`

Change Cadence Target Assignee:

`/services/data/vXX.X/actions/standard/changeSalesCadenceTargetAssignee`

Modify Cadence Tracker Attributes:

`/services/data/vXX.X/actions/standard/modifyCadenceTrackerAttributes`

Send Cadence Event:

`/services/data/vXX.X/actions/standard/sendSalesCadenceEvent`

Select Template For Cadence Step Tracker:

`/services/data/vXX.X/actions/standard/selectTemplateForSalesCadenceStepTracker`

Formats

JSON, XML

HTTP Methods

GET, HEAD, POST

Authentication

`Authorization: Bearer token`

Other Information

Error Response Types on page 88

Assign Target to Cadence

Available in API version 45.0.

URI: `/services/data/vXX.X/actions/standard/assignTargetToSalesCadence`

Table 11: Inputs

Input	Details
`salesCadenceNameOrId`	**Type** string **Description** Required. The name or ID of the cadence.
`startStepNameOrId`	**Type** string **Description** The name or ID of the cadence step where the target starts in the cadence.
`targetId`	**Type** ID **Description** Required. The ID of the contact, a lead, or person account to add to the cadence.
`userId`	**Type** ID **Description** The ID of the user designated as the target assignee. The target assignee is the sales rep who performs the cadence steps for the target.
`relatedToId`	**Type** ID **Description** The ID of the target's related opportunity or invoice. This field is only available when Relate Opportunities to Cadences or Use Cadences for Collections is turned on in Sales Engagement Setup.

Sample Input

The following code sample adds a target to a cadence:

```
{
  "inputs" : [ {
    "salesCadenceNameOrId" : "77Cxx0000004CXEEA2",
    "targetId" : "00Qxx000002TRI2EAO"
    }]
}
```

Remove Target from Cadence

Available in API version 45.0.

URI: `/services/data/vXX.X/actions/standard/removeTargetFromSalesCadence`

Table 12: Inputs

Input	Details
`targetId`	**Type** ID **Description** Required if `actionCadenceTrackerId` is null. The ID of the contact, a lead, or person account to remove from the cadence.
`actionCadenceTrackerId`	**Type** ID **Description** Required if `targetId` is null. The ID of the action cadence tracker to remove from the cadence.
`completionReasonCode`	**Type** string **Description** Required. The completion reason code indicates how the target competed the cadence. Valid value is: • ManuallyRemoved
`completionDisposition`	**Type** string **Description** The disposition of the completed cadence tracker. Valid values are: • Success • Customer Engaged • Customer Connected • Contact Later • No Response • Not Interested • Disqualified • Bad Data • Duplicate
`relatedToId`	**Type** ID **Description** The ID of the target's related opportunity or invoice. This field is only available when Relate Opportunities to Cadences or Use Cadences for Collections is turned on in Sales Engagement Setup.

Input	Details
`relatedToAttributionType`	**Type** string **Description** The attribution type of the target's related opportunity or invoice. This field is only available when Relate Opportunities to Cadences or Use Cadences for Collections is turned on in Sales Engagement Setup. Valid values are: • Activation (Valid for both opportunities and invoices.) • Maturation (Valid for opportunities.) • Collected (Valid for invoices.)
`shouldApplyUserContext`	**Type** boolean **Description** Indicates whether to remove only action cadence trackers owned by the running user (`true`) or not (`false`).

Sample Input

The following code sample removes a target from a cadence:

```
{
  "inputs" : [ {
    "completionReasonCode" : "ManuallyRemoved",
    "targetId" : "00Qxx000002TRI2EAO"
    }]
}
```

Pause Cadence Tracker

Pause a target in its cadence. Available in API version 50.0.

URI: `/services/data/vXX.X/actions/standard/pauseSalesCadenceTracker`

Table 13: Inputs

Input	Details
`targetId`	**Type** ID **Description** Required. The ID of the contact, a lead, or person account to pause.
`resumeTime`	**Type** String **Description** Optional. The scheduled end time for the pause.

Sample Input

The following code sample pauses a target in its cadence:

```
{
  "inputs" : [ {
    "targetId" : "00Qxx000002TRI2EAO", "resumeTime" : "2021-06-15T05:30:00:521917Z"
    }]
}
```

Resume Cadence Tracker

Resume a target in its cadence. Available in API version 50.0.

URI: `/services/data/vXX.X/actions/standard/resumeSalesCadenceTracker`

Table 14: Inputs

Input	Details
`targetId`	**Type** ID **Description** Required. The ID of the contact, a lead, or person account to pause.

Sample Input

The following code sample resumes a target in its cadence:

```
{
  "inputs" : [ {
    "targetId" : "00Qxx000002TRI2EAO"
    }]
}
```

Change Cadence Target Assignee

Available in API version 50.0.

URI: `/services/data/vXX.X/actions/standard/changeSalesCadenceTargetAssignee`

Table 15: Inputs

Input	Details
`targetId`	**Type** ID **Description** Required. The ID of the contact, a lead, or person account to pause.
`userId`	**Type** ID

Input	Details
	Description The ID of the user designated as the target assignee. The target assignee is the sales rep who performs the cadence steps for the target.

Sample Input

The following code changes a target's assignee:

```
{
  "inputs" : [ {
    "targetId" : "00Qxx000002TRI2EAO",
    "userId" : "005R0000000eg3zIAA",
    }]
}
```

Modify Cadence Tracker Attributes

Available in API version 51.0.

URI: `/services/data/vXX.X/actions/standard/modifyCadenceTrackerAttributes`

Table 16: Inputs

Input	Details
`actionCadenceTrackerId`	**Type** ID **Description** Required. The ID of the cadence tracker to modify.
`completionDisposition`	**Type** string **Description** The disposition of the completed cadence tracker. Valid values are: • Success • Customer Engaged • Customer Connected • Contact Later • No Response • Not Interested • Disqualified • Bad Data • Duplicate

Input	Details
relatedToId	**Type** ID **Description** The ID of the target's related opportunity or invoice. This field is only available when Relate Opportunities to Cadences or Use Cadences for Collections is turned on in Sales Engagement Setup.
relatedToAttributionType	**Type** string **Description** The attribution type of the target's related opportunity or invoice. This field is only available when Relate Opportunities to Cadences or Use Cadences for Collections is turned on in Sales Engagement Setup. Valid values are: • Activation (Valid for both opportunities and invoices.) • Maturation (Valid for opportunities.) • Collected (Valid for invoices.)

Sample Input

The following code modifies a cadence tracker with a Completion Disposition of "Customer Engaged", a related opportunity, and an Attribution Type of "Activation":

```
{
  "inputs" : [ {
    "actionCadenceTrackerId" : "0qBR00000005CXvMAM",
    "completionDisposition" : "Customer Engaged",
    "relatedToId" : "006R0000003DNpJIAW",
    "relatedToAttributionType" : "Activation"
    }]
}
```

Send Cadence Event

Available in API version 52.0.

Send an event to a cadence, such as skipping or manually completing a step.

URI: `/services/data/vXX.X/actions/standard/sendSalesCadenceEvent`

Table 17: Inputs

Input	Details
recordId	**Type** ID **Description** Required. The ID of the cadence step tracker to send the event to.

Input	Details
eventType	**Type** string **Description** Required. The type of event to send. Valid values are: • Skip • Manual Complete

Sample Input

The following code sends a Manual Complete event to a cadence step tracker:

```
{
  "inputs" : [ {
    "recordId" : "8HFR00000005WyqOAE",
    "eventType" : "Manual Complete"
    }]
}
```

Select Template for Cadence Step Tracker

Retrieve the email template or call script from a cadence step or cadence step variant (if variant testing) to be used while executing a step for a particular target. Available in API version 53.0.

URI: `/services/data/vXX.X/actions/standard/selectTemplateForSalesCadenceStepTracker`

Table 18: Inputs

Input	Details
stepTrackerId	**Type** ID **Description** Required. The ID of the cadence step tracker.

Table 19: Outputs

Output	Details
output	**Type** JSON **Description** The email template or call script ID and its related split percentage.
error	**Type** string

Output	Details
	Description The error message returned when the action fails.

Sample Input

The following code retrieves the email template or call script for two cadence steps:

```
{"inputs": [ {
        "stepTrackerId" : "8HFR00000006LE8OAM"
    },
    {
        "stepTrackerId" : "8HFR00000006LEDOA2"
    }]
}
```

Sample Output

The following code sample illustrates a response after one success and one failure.

```
[ {
  "actionName" : "selectTemplateForSalesCadenceStepTracker",
  "errors" : null,
  "isSuccess" : true,
  "outputValues" : {
    "output" : {
      "SplitPercentage" : 10.0,
      "TemplateId" : "00XR0000000UOtZMAW"
    }
  }
}, {
  "actionName" : "selectTemplateForSalesCadenceStepTracker",
  "errors" : [ {
    "statusCode" : "UNKNOWN_EXCEPTION",
    "message" : "No template was found.",
    "fields" : [ ]
  } ],
  "isSuccess" : false,
  "outputValues" : {
    "error" : "No template was found."
  }
} ]
```

Error Response Types

Sales Engagement actions can respond with success or errors.

If any type of error occurs with an action, the `isSuccess` field is `false`.

This example illustrates a success response for the Assign Target To Cadence action.

```
[ {
         "actionName" : "assignTargetToSalesCadence",
```

```
        "isSuccess" : true
        } ]
```

This example illustrates an error caused by sending invalid input values to the Assign Target To Cadence action.


```
[ {
        "actionName" : "assignTargetToSalesCadence",
        "errors" : [ {
        "statusCode" : "UNKNOWN_EXCEPTION",
        "message" : "The object needs to be a valid cadence entity.",
        "fields" : [ ]
        } ],
        "isSuccess" : false,
        "outputValues" : {
        "error" : "The object needs to be a valid cadence entity."
        }
        } ]
```


Salesforce Omnichannel Inventory Actions

Manage inventory availability and provide omnichannel commerce experiences in flows with Salesforce Omnichannel Inventory.

For more information about using Omnichannel Inventory actions in flows, see Salesforce Omnichannel Inventory Flow Core Actions in Salesforce Help.

These actions are available in API version 51.0 and later.

Your org must have Salesforce Omnichannel Inventory enabled.

Supported REST HTTP Methods

URI

Get a specific Omnichannel Inventory action:

`/services/data/vXX.X/actions/standard/oci_action_name`

Formats

JSON, XML

HTTP Methods

GET

Authentication

`Authorization: Bearer token`

Notes

You can also call the corresponding Connect REST API endpoints or Apex ConnectApi methods. For more information, see Omnichannel Inventory Resources in the *Connect REST API Developer Guide* and ConnectApi Namespace in the *Apex Developer Guide*.

In flows, Omnichannel Inventory action inputs and outputs use Apex-defined variables that map to input and output classes in the ConnectApi namespace.

Salesforce Order Management Actions

Manage, fulfill, and service orders in flows with Salesforce Order Management.

For more information about using Order Management actions in flows, see Salesforce Order Management Flow Core Actions in Salesforce Help.

These actions are available in API version 48.0 and later.

Your org must have a Salesforce Order Management license.

Supported REST HTTP Methods

URI

Get a specific Order Management action:

`/services/data/vXX.X/actions/standard/om_action_name`

Formats

JSON, XML

HTTP Methods

GET

Authentication

`Authorization: Bearer token`

Notes

You can also call the corresponding Connect REST API endpoints or Apex ConnectApi methods. For more information, see Order Management Resources in the *Connect REST API Developer Guide* and ConnectApi Namespace in the *Apex Developer Guide*.

In flows, Order Management action inputs and outputs use Apex-defined variables that map to input and output classes in the ConnectApi namespace.

Send Conversation Messages Actions

Send a messaging component to users in enhanced WhatsApp, enhanced Apple Messages for Business, enhanced SMS, or Messaging for In-App when the targeted channel has bandwidth. Each Send Conversation Messages action corresponds to a supported messaging component.

This object is available in API version 59.0 and later.

On invocation, this action inserts and enqueues a message for handling the request and sending out the messages async. The request record can be used to track the progress.

Typical use cases include:

- Confirmation of a purchase or booking
- Shipping or delivery updates
- Password reset requests
- Account verification messages
- Payment reminders
- Abandoned cart reminders

Messages are sent immediately as long as the following conditions are met. If these conditions aren't met, messages can be queued for sending, resulting in a slight delay.

- The invocation of the Send Conversation Messages action includes 5 or fewer messages. If it includes more, the additional messages are queued.

- No more than 200 invocations of the Send Conversation Messages action are in progress. If this limit is reached, additional requests are queued and sent when the number of in-progress requests falls below 200. For queued requests, the messaging session owner for automated messages is the Platform Integration User. Otherwise, it's the user triggering the action.
- The conversation platform has capacity available to send the message. Messages in active conversations are always prioritized over automated outbound messages.

Supported REST HTTP Methods

URI

`/services/data/v`***`59.0`***`/actions/standard/sendConversationMessages`

Formats

JSON, XML

HTTP Methods

GET, HEAD, POST

Authentication

`Authorization: Bearer` ***`token`***

Inputs

Input	Details
`allowedSessionStatus`	**Type** string **Description** Limits the time when the message can be sent. Valid values are: • `Any`—Send the message whether the messaging user is engaged in a messaging session with the business or not. • `NonActive`—Send the message unless the messaging user is engaged in a messaging session with a status of Active. The message is sent after the session's status changes. • `Closed`—Send the message unless the messaging user is engaged in a messaging session. The message is sent after the session is closed.
`channelConsentType`	**Type** string **Description** How to apply consent preferences when determining which messaging users receive the message. Valid values are: • `MessagingEndUser`—Apply messaging users' consent settings for a channel. This is the most common approach. • `CommunicationSubscription`— Send the message only to messaging end users who have opted into a particular subscription. When selected, a `communicationSubscriptionId` must also be provided. This option is visible only if you have a unified channel that supports marketing interactions.

Input	Details
	• `Custom`—Apply custom consent preferences. Used if `isEnforceMessagingChannelConsent` is set to `false`.
communicationSubscriptionId	**Type** string **Description** (Available if `channelConsentType` is set to `CommunicationSubscription`) The subscription that controls which messaging users the message is sent to. The subscription must be tied to the channel where the message is sent, and the message is sent only to users who opted in to the subscription.
isEnforceMessagingChannelConsent	**Type** boolean **Description** Indicates whether messaging consent must be verified after messages are queued for sending.
messageDefinitionName	**Type** string **Description** The API name of a ConversationMessageDefinition (messaging component) record that's used to render the messages.
messagingDefinitionInputParameters	**Type** list **Description** Optional. A collection of Apex `richmessaging__MessageDefinitionInputParameter` records that contain messaging component details to use when rendering messages.
messagingEndUserIds	**Type** list **Description** A collection of up to 100 messaging end user record IDs to use as recipients of the messages. To send more than 100 messages, divide your end user recipients into batches of 100 or fewer and repeat the action invocation for each batch.
requestType	**Type** string **Description** Specifies the type of the request. Valid values are: `SendNotificationMessages`.

Outputs

Input	Details
`requestId`	**Type** string **Description** The ID of the ConvMessageSendActionRequest record created by the request that's used to track the message progress.
`messageIdentifiers`	**Type** list **Description** A collection of generated message UUIDs with one entry for each recipient specified in messagingUserIds.

Usage

Sample Input

The following sample input attempts to create a ConvMessageSendRequest record using a Messaging Component (`messageDefinitionName`), the request for the type of component being sent (`requestType`), the consent preferences (`isEnforceMessagingChannelConsent`), the consent type (`channelConsentType`), when the message can be sent (`allowedSessionStatus`), and the message recipients (`messagingEndUserIds`). Additionally, the request contains a list of custom parameters (`messagingDefinitionInputParameters`). These parameters are mapped to parameters configured in the messaging component, which can be optional. Applicable data types for parameters are `textValue`, `recordIdValue`, `dateValue`, `dateTimeValue`, `numberValue`, and `booleanValue`.

```
{
  "inputs": [{
      "messageDefinitionName":"Notification",
      "requestType":"SendNotificationMessages",
      "isEnforceMessagingChannelConsent":true,
      "channelConsentType":"MessagingEndUser",
      "allowedSessionStatus":"Any",
      "messagingEndUserIds":"0PARM000000Lc3I,0PARM000000MZ3p",

"messagingDefinitionInputParameters":[{"name":"custom_parameter_name","textValue":"custom
 parameter value"}]
  }]
}
```

Sample Output

The following sample output illustrates a response after a successful request.

```
[
  {
    "actionName":"sendConversationMessages",
    "errors":null,
```

```
    "isSuccess":true,
    "outputValues":{

"messageIdentifiers":"c581098c-5db6-4ed8-915f-c9aaa016c671,d8e1990e-5d67-414c-9713-180107d7d1bb",
"requestId":"1srM000000000p"
    }
  }
]
```


For more information about this action, see Flow Core Actions: Send Conversation Messages and Send Automated Messages in Enhanced Messaging Channels in Salesforce Help.

Send Notification Actions

Call a notification type to send. Each Send Notification action corresponds to a supported notification type. This object is available in API version 54.0 and later. Send Notification actions are available only for Slack-enabled custom notification types and certain Slack-enabled standard notification types.

Note: To send notifications for Slack, enable Salesforce for Slack Integrations.

To create a custom Slack notification type supported by a Send Notification action, see Create and Send Custom Slack Notifications.

To trigger Send Notification actions using REST API calls, you need the Send Custom Notifications user permission.

Supported REST HTTP Methods

URI

Get a list of available Send Notification actions.

`/services/data/vXX.X/actions/custom/sendNotification`

Get information about a specific Send Notification action:

`/services/data/vXX.X/actions/custom/sendNotification/notification_type_name`

Formats

JSON, XML

HTTP Methods

GET, HEAD, POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`recipientIds`	**Type** ID **Description** Required. The IDs of the notification recipients or recipient types. Valid recipient or recipient type values are UserId or CollaborationRoomId values.

Input	Details
`recordId`	**Type** ID **Description** Required. The ID of the record that the notifications are about. The record ID must be associated with the specific EntityType of the notification type. For example, enter the record ID for an opportunity when configuring a notification type associated with the Opportunity object. For custom notification types, you can find the related object by viewing the notification type's settings from Custom Notifications in Setup. For supported standard notification types, refer to the Standard Notification Types and Related Objects table.

Standard Notification Types and Related Objects

Use this table to identify which object applies to each standard notification type supported by a Send Notification action. The object determines the value that you enter for `recordId`.

Standard Notification Type	Related Salesforce Object
`amount_updated`	Opportunity
`close_date_reminder`	Opportunity
`close_date_updated`	Opportunity
`deal_won`	Opportunity
`deals_to_watch`	Opportunity
`hc_allergy_intolerance_alert`	Allergy Intolerance
`hc_care_determinant_alert`	Care Determinant
`hc_care_plan_alert`	Case
`hc_care_plan_task_alert`	Task
`hc_health_condition_alert`	Health Condition
`high_priority_case`	Case
`new_child_opportunity`	Opportunity
`next_step_reminder`	Opportunity

Standard Notification Type	Related Salesforce Object
`stage_reminder`	Opportunity
`stage_updated`	Opportunity

SEE ALSO:

Object Reference for the Salesforce Platform: CollaborationRoom

Object Reference for the Salesforce Platform: Swarm

Session-Based Permission Set Actions

Activate or deactivate a session-based permission set for the current user's API session.

This object is available in API version 40.0 and later.

This action activates or deactivates the provided permission set for the current user's API session. The activation or deactivation doesn't affect other sessions. The permission set must already be assigned to the current user.

Supported REST HTTP Methods

URI

Activate session-based permission set:

`/services/data/vXX.X/actions/standard/activateSessionPermSet`

Deactivate session-based permission set:

`/services/data/vXX.X/actions/standard/deactivateSessionPermSet`

Formats

JSON, XML

HTTP Methods

GET, HEAD, POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`PermSetName`	**Type** string **Description** Required. Specifies the developer name of the permission set.

Input	Details
`PermSetNamespace`	**Type** string **Description** Optional. Specifies the namespace of the permission set.

Outputs

None.

Simple Email Actions

Send an email where you specify the subject, body, and recipients. Available in API version 32.0 and later.

Email Sending Limits

If you're using `logEmailOnSend` or `emailTemplateId`, the daily email-sending limit is based on the single email limit. See General Email Limits.

If you're not using `logEmailOnSend` or `emailTemplateId`, the daily email-sending limit is based on the daily workflow email limit. See Proactive Alert Monitoring: Daily Workflow Email Limit.

Supported REST HTTP Methods

URI

Get a list of available simple email actions:

`/services/data/vXX.X/actions/standard/emailSimple`

Formats

JSON, XML

HTTP Methods

GET, HEAD, POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
`addThreadingTokenToBody`	**Type** boolean **Description** Optional. Indicates whether to create a unique token for the related record and add it to the email body.

Input	Details
	When the related record is a case record, Email-to-Case uses the token to link future email responses to that case. To link future email responses to other records, create an Apex Email Service and use the `EmailMessages.getRecordIdFromEmail` function to find the record that matches the token.
`addThreadingTokenToSubject`	**Type** boolean **Description** Optional. Indicates whether to create a unique token for the related record and add it to the email subject. When the related record is a case record, Email-to-Case uses the token to link future email responses to that case. To link future email responses to other records, create an Apex Email Service and use the `EmailMessages.getRecordIdFromEmail` function to find the record that matches the token.
`attachmentId`	**Type** string **Description** Optional. A comma-delimited list of attachment IDs in the email. This parameter accepts single-value resources of the string type that contain a list of attachments. The value is treated as text. This field is available in API version 63.0 and later. The attachment ID can be of a Document, Content Version, or Attachment items.
`bccRecipientAddressList`	**Type** text **Description** Optional. A comma-delimited list of recipient email addresses to send a copy of the email to. Email addresses in the BCC list are hidden from all recipients. This parameter accepts single-value resources of any type. The maximum size for this field is 4,000 bytes. This field is available in API version 62.0 and later. If `ccRecipientAddressList`, `emailAddresses`, `emailAddressesArray`, and `recipientId` are also used, the combined number of recipients must be 150 or fewer.
`ccRecipientAddressList`	**Type** text **Description** Optional. A comma-delimited list of recipient email addresses to send a copy of the email to. This parameter accepts single-value resources of any type. The maximum size for this field is 4,000 bytes. This field is available in API version 62.0 and later.

Input	Details
	If `bccRecipientAddressList`, `emailAddresses`, `emailAddressesArray`, and `recipientId` are also used, the combined number of recipients must be 150 or fewer.
`emailAddresses`	**Type** string **Description** Optional. A comma-delimited list of the recipients' email addresses. The maximum size for this field is 4,000 bytes. If `bccRecipientAddressList`, `ccRecipientAddressList`, `emailAddressesArray`, and `recipientId` are also used, the combined number of recipients must be 150 or fewer.
`emailAddressesArray`	**Type** string **Description** Optional. A collection of the recipients' email addresses. Up to five email recipients, specified as a collection of strings. The maximum size for this field is 4,000 bytes. If `bccRecipientAddressList`, `ccRecipientAddressList`, `emailAddresses`, and `recipientId` are also used, the combined number of recipients must be 150 or fewer.
`emailBody`	**Type** textarea **Description** Optional. The body of the email. Required if you're not using an email template.
`emailSubject`	**Type** string **Description** Optional. The subject of the email. Required if you're not using an email template. The value is treated as plain text.
`emailTemplateId`	**Type** text **Description** Optional. The ID of the email template to use for the subject and body of the email. If this input is included, `recipientId` is required. If the email template has merge fields from an object other than the one associated with `recipientId`, specify the record used to supply those merge fields in `relatedRecordId`. This field is available in API version 58.0 and later. If this input is specified, it changes the API called by the action, which can impact your daily email-sending limit. See Flow Core Action: Send Email in Salesforce Help.
`logEmailOnSend`	**Type** boolean

Input	Details
	Description Optional. Indicates whether to log the email on the specified records' activity timelines. Valid values are `true` and `false`. Default value is `false`, and the email isn't logged. To log an email, you must specify a value in `recipientId` or `relatedRecordId`. This field is available in API version 58.0 and later. If this input is set to `true`, it changes the API called by the action, which can impact your daily email-sending limit. See Flow Core Action: Send Email in Salesforce Help.
`recipientId`	**Type** text **Description** Optional. The ID of a lead, a contact, or a user record. If `logEmailOnSend` is included, then `recipientId` is the ID of the person to send and log the email to. If `emailTemplateId` is included, then `recipientId` is required and is the ID of the person to send the email to. The maximum size for this field is 4,000 bytes. This field is available in API version 58.0 and later. If `bccRecipientAddressList`, `ccRecipientAddressList`, `emailAddresses`, and `emailAddressesArray` are also used, the combined number of recipients must be 150 or fewer.
`relatedRecordId`	**Type** text **Description** Optional. The ID of a record that's not a person. For example, the ID of a case record. If `logEmailOnSend` is included, `relatedRecordId` is the ID of a secondary record to log the email to. In this case, `relatedRecordId` can't be used to log an email if `recipientId` is a lead record. This field is available in API version 58.0 and later. If `emailTemplateId` is included, `relatedRecordId` is the ID of the non-recipient record used to populate email template merge fields.
`sendRichBody`	**Type** boolean **Description** Optional. Indicates whether the body of the email uses rich or plain text. Valid values are: • *`True`*—The body of the email uses rich text. • *`False`*—The body of the email uses plain text. This value is the default.
`senderAddress`	**Type** string **Description** Optional. The organization-wide email address to be used as the sender. Required only if `senderType` is set to `OrgWideEmailAddress`.

Input	Details
	If a scheduled flow sets `senderType` to `OrgWideEmailAddress`, then `senderAddress` is required.
`senderType`	**Type** string **Description** Optional. Email address used as the email's From and Reply-To addresses. Valid values are: • `CurrentUser`—Email address of the user running the flow. This setting is the default. • `DefaultWorkflowUser`—Email address of the default workflow user. • `OrgWideEmailAddress`—The organization-wide email address that is specified in `senderAddress`. In scheduled flows, the only valid value is `OrgWideEmailAddress`.
`Use Line Breaks`	**Type** boolean **Description** Optional. Indicates whether to render the line breaks in the rich-text-formatted body text template. Valid values are `true` and `false`. The default value is `false`.

Outputs

None.

Submit Exchange Order

Submits an exchange order based on the specified information.

To access, you need the following permissions.

- Salesforce Order Management License or Salesforce B2B Commerce License

This object is available in API version 60.0 and later.

Supported REST HTTP Methods

URI

`/services/data/v59.0/actions/standard/previewCartToExchangeOrder`

Formats

JSON, XML

HTTP Methods

GET, HEAD, POST

Authentication

Authorization: Bearer token

Inputs

Input	Details
`exchangeCartId`	**Type** ID **Description** Required. The ID of the cart record containing the items in the exchange order.
`orderNumber`	**Type** ID **Description** Optional. Order number for the created exchange order summary.
`orderSummaryId`	**Type** ID **Description** Required. The ID of the order summary record associated with the list of exchanges.
`paymentInfoInputs`	**Type** Collection **Description** Optional. A collection of Apex ConnectApi.PaymentInfoInputRepresentation records, each containing payment details when the exchange order amount is greater than the original order amount.
`referenceId`	**Type** ID **Description** Required. The ID of the record that's related to the specified order summary. Only IDs from a return order record are supported.
`sequenceList`	**Type** Collection **Description** Optional. A collection of Apex ConnectApi.SequenceOrderPaymentSummaryInputRepresentation records to reserve a balance from. Each record contains an order summary payment ID and an amount.

Outputs

Output	Details
`changeBalances`	**Type** string **Description** A string that contains the calculated amounts resulting from the exchange order.
`errors`	**Type** string **Description** Following a `400` error response, the error objects show information about the error that occurred. Contains a status code, message, and list of fields.
`exchangeOrderSummaryId`	**Type** ID **Description** The ID of the order summary record associated with the list of exchanges.
`isSuccess`	**Type** boolean **Description** The ID of the order summary record created for the exchanges.
`orderSummaryID`	**Type** ID **Description** The ID of the order summary record associated with the list of exchanges.

Submit for Approval Actions

Submit a Salesforce record for approval if an approval process is defined for the current entity.

For more information about creating submit for approval actions, see Creating Approval Actions, and to learn more about approval processes, see Approval Process Overview, in Salesforce Help.

This object is available in API version 32.0 and later.

Supported REST HTTP Methods

URI

Get a list of actions:

`/services/data/vXX.X/actions/standard/submit`

Formats
JSON, XML

HTTP Methods
GET, HEAD, POST

Authentication
`Authorization: Bearer` ***token***

Inputs

Input	Details
`comment`	**Type** string **Description** Optional. Comments that accompany the Approval submission.
`nextApproverIds`	**Type** reference **Description** Optional. An array of one ID of the next approver, which can be a user or a group. This input is optional because some approval processes have the next approver specified in the approval process definition. You can't specify more than 2,000 approvers.
`objectId`	**Type** reference **Description** Required. The ID of the record being submitted for approval.
`processDefinitionNameOrId`	**Type** string **Description** Optional. The ID or name of the approval process to run against the record. If none is specified, the first approval process whose entry criteria the record satisfies is used. Names can't be longer than 120 characters.
`skipEntryCriteria`	**Type** boolean **Description** Optional. A Boolean value controlling whether the entry criteria for the specified approval process must be evaluated for the record (`true`) or not (`false`). If set to true, also specify `processDefinitionNameOrId`.
`submitterId`	**Type** string

Input	Details
	Description Optional. The ID of the user submitting the record for approval. If none is specified, the submitter is the current user.

Outputs

Output	Details
`actorIds`	**Type** reference **Description** An array of the IDs of the next approvers.
`entityId`	**Type** reference **Description** The ID of the record submitted for approval.
`instanceId`	**Type** reference **Description** The ID of the approval process instance.
`instanceStatus`	**Type** string **Description** The status of the approval. The valid values are • Approved • Pending • Rejected • Removed
`newWorkItemIds`	**Type** reference **Description** An array of the IDs of the work items created for the next step in this approval process.

Survey Invitation Actions

Send email survey invitations to leads, contacts, and users in your org based on an action. Also, send customized notifications to users about important events or updates to the records that they're working on.

Dynamic Send Survey Invitation Actions

Send customized notifications to users about important events or updates to the records that they're working on. For example, notify account owners when a case is created.

This action is available in API version 51.0 and later.

Special Access Rules

To access the Dynamic Send Survey Invitation action, you must have the Feedback Management Survey Response Pack and the Salesforce org enabled with Surveys.

Supported REST HTTP Methods

URI

Get the list of invocable actions for each available survey.

`/services/data/v51.0/actions/custom/dynamicSendSurveyInvitation`

Send survey invitation by email by using the invocable action.

`/services/data/v51.0/actions/custom/dynamicSendSurveyInvitation/{surveyName}`

Formats

JSON, XML

HTTP Methods

GET, HEAD, POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
anonymousResponse	**Type** boolean **Description** Indicates whether the participant's name is recorded in the response record (`true`) or not (`false`).
autoExpiryDays	**Type** integer

Input	Details
	Description Number of days remaining for the survey invitation to expire.
`emailTemplateName`	**Type** string **Description** Developer name of the template that contains either the question or the survey link.
`isUnauthenticatedResponse`	**Type** boolean **Description** Indicates whether the participant is required to authenticate before starting the survey (`true`) or not (`false`).
`personalInvitation`	**Type** boolean **Description** Indicates whether the invitation is specific to the recipient (`true`) or not (`false`).
`recipient`	**Type** reference **Description** Required. Salesforce ID of the record that the survey invitation is sent to.
`recipientType`	**Type** string **Description** Type of the survey recipient.
`surveyQuestionName`	**Type** string **Description** Developer name of the question that's sent by using the invitation.
`surveySubjectEntity`	**Type** reference **Description** ID of the record that associates the invitation record with another record.

Outputs

None.

Example

GET

This example shows how to get information about the Dynamic Send Survey Invitation action type.

```
curl --include --request GET \
--header "Authorization: Authorization: Bearer 00DR...xyz" \
--header "Content-Type: application/json" \
"https://instance.salesforce.com/services/data/v51.0/actions/custom/dynamicSendSurveyInvitation"
```

Here’s a response that returns the list of invocable actions for each survey.

```
{
  "actions" : [ {
    "label" : "flowsend",
    "name" : "flowsend",
    "type" : "SEND_SURVEY_DYNAMIC_INVOCABLE_ACTION",
    "url" : "/services/data/v51.0/actions/custom/dynamicSendSurveyInvitation/flowsend"
  }, {
    "label" : "survey2",
    "name" : "survey2",
    "type" : "SEND_SURVEY_DYNAMIC_INVOCABLE_ACTION",
    "url" : "/services/data/v51.0/actions/custom/dynamicSendSurveyInvitation/survey2"
  }, {
    "label" : "survey",
    "name" : "survey",
    "type" : "SEND_SURVEY_DYNAMIC_INVOCABLE_ACTION",
    "url" : "/services/data/v51.0/actions/custom/dynamicSendSurveyInvitation/survey"
  } ]
}
```

POST

Here’s a request for the Dynamic Send Survey Invitation action.

```
{
   "inputs":[{
      "recipient" : "003xx000004WpemAAC",
      “isUnauthenticatedResponse” : false,
      "autoExpiryDays" : 6
   }]
}
```

Here’s a response for the Dynamic Send Survey Invitation action.

```
[
   {
      "actionName" : "survey",
      "errors" : null,
      "isSuccess" : true,
      "outputValues" : null
```

```
    }
]
```

SEE ALSO:

Salesforce Help: Send Survey Invitations Using Flows

Send Survey Invitation Actions

Send email survey invitations to leads, contacts, and users in your org based on an action. For example, send a survey invitation when a customer support case closes.

This action is available in API version 47.0 and later.

Special Access Rules

To access the Send Survey Invitation action, you must have the Feedback Management Survey Response Pack and the Salesforce org enabled with Surveys.

Supported REST HTTP Methods

URI

`/services/data/v47.0/actions/standard/sendSurveyInvitation`

Formats

JSON, XML

HTTP Methods

GET, HEAD, POST

Authentication

`Authorization: Bearer token`

Inputs

Input	Details
anonymousResponse	**Type** boolean **Description** Indicates whether the participant's name is recorded in the response record (`true`) or not (`false`).
autoExpiryDays	**Type** integer **Description** Number of days remaining for the survey invitation to expire.

Input	Details
emailTemplateName	**Type** string **Description** Developer name of the template that contains either the question or the survey link.
isUnauthenticatedResponse	**Type** boolean **Description** Indicates whether the participant is required to authenticate before starting the survey (`true`) or not (`false`).
personalInvitation	**Type** boolean **Description** Indicates whether the invitation is specific to the recipient (`true`) or not (`false`).
recipient	**Type** reference **Description** Required. Salesforce ID of the record that the survey invitation is sent to. The record can be a user (internal invitation) or a contact or a lead (external invitation via community).
surveyName	**Type** string **Description** Required. Developer name of the survey that the invitation is sent for.
surveyQuestionName	**Type** string **Description** Developer name of the question that’s sent using the invitation.
surveySubjectEntity	**Type** reference **Description** ID of the record that associates the invitation record with another record.

Outputs

None.

Example

GET

This example shows how to get information about the Send Survey Invitation action type.

```
curl --include --request GET \
--header "Authorization: Authorization: Bearer 00DR...xyz" \
--header "Content-Type: application/json" \
"https://instance.salesforce.com/services/data/v47.0/actions/standard/sendSurveyInvitation"
```

POST

Here's a request for the send survey invitation action.

```
{
   "inputs":[{
      "surveyName" : "FlowActionSend",
      "recipient"  : "003RO0000037IRvYAM",
      "autoExpiryDays" : 6
   }]
}
```

Here's a response for the send survey invitation action.

```
[
   {
      "actionName" : "sendSurveyInvitation",
      "errors" : null,
      "isSuccess" : true,
      "outputValues" : null
   }
]
```

SEE ALSO:

Salesforce Help: Send Survey Invitations Using Process Builder

Work Plan and Work Step Actions

Manage work plans and work steps using actions.

For more information about Field Service, see the Field Service Developer Guide.

These actions are available in API version 52.0 and later. They require Field Service to be enabled.

Add Work Plans Limits

You can generate work plans linked to work orders via the `addWorkPlans` flow, but, by default, users can only generate 100 work plans per work order.

Supported REST HTTP Methods

URIs

Add work plans: `/services/data/vXX.X/actions/standard/addWorkPlans`

Add work steps: `/services/data/vXX.X/actions/standard/addWorkSteps`

Generate work plans: `/services/data/vXX.X/actions/standard/generateWorkPlans`

Delete work plans: `/services/data/vXX.X/actions/standard/deleteWorkPlans`

Formats

JSON, XML

HTTP Methods

GET, HEAD, POST

Authentication

`Authorization: Bearer token`

Add Work Plans

URI: `/services/data/vXX.X/actions/standard/addWorkPlans`

This action creates work plan records from the work plan library.

Table 20: Inputs

Input	Details
`recordId`	**Type** string **Description** Required. The ID of the work order or work order line item to add the work plans to.
`workPlanTemplateIdList`	**Type** array of strings **Description** Required. The IDs of the work plan templates used to instantiate the work plans.

Table 21: Outputs

Output	Details
`recordId`	**Type** string **Description** The ID of the work order or work order line item.
`workPlanIdList`	**Type** array of strings **Description** The list of work plan IDs.

Sample Input

The following code sample adds work plans:

```
{
    "inputs" : [ {
        "recordId" : "0WOxx00000007j3",
        "workPlanTemplateIdList" : ["7Iyxx0000004LSS", "7Iyxx0000004LTT"]
    }]
}
```

Sample Output

The following code sample illustrates a response after a successful request.

```
[ {
    "actionName" : "addWorkPlans",
    "errors" : null,
    "isSuccess" : true,
    "outputValues" : {
        "recordId" : "0WOxx00000007j3",
        "workPlanIdList" : ["0gqxx0000000Adh", "0gqxx0000000Adi"]
    }
} ]
```

Add Work Steps

URI: `/services/data/vXX.X/actions/standard/addWorkSteps`

This action creates work step records from the work plan library.

Table 22: Inputs

Input	Details
`recordId`	**Type** string **Description** Required. The ID of the work plan to add the work steps to.
`workStepTemplateIdList`	**Type** array of strings **Description** Required. The IDs of the work step templates used to instantiate the work steps.

Table 23: Outputs

Output	Details
`recordId`	**Type** string

Output	Details
	Description The ID of the work order or work order line item.
`workStepIdList`	**Type** array of strings **Description** The list of work step IDs.

Sample Input

The following code sample adds work steps:

```
{
    "inputs" : [ {
        "recordId" : "0gqRM0000004DxoYAE",
        "workStepTemplateIdList" : ["4L0xx0000004FJoCAM", "4L0xx0000004FJoNAC"]
    }]
}
```

Sample Output

The following code sample illustrates a response after a successful request.

```
[ {
    "actionName" : "addWorkSteps",
    "errors" : null,
    "isSuccess" : true,
    "outputValues" : {
       "recordId" : "0gqRM0000004DxoYAE",
       "workstepIdList" : ["0hFxx00000007uLEAQ", "0hFxx00000007uRAUW"]
    }
} ]
```

Generate Work Plans

URI: `/services/data/vXX.X/actions/standard/generateWorkPlans`

This action generates work plans based off rules defined in the work plan library.

Table 24: Inputs

Input	Details
`recordId`	**Type** string **Description** Required. The ID of the work order or work order line item to generate work plans for.

Table 25: Outputs

Output	Details
`recordId`	**Type** string **Description** The ID of the work order or work order line item.
`workPlanIdList`	**Type** array of strings **Description** The list of work plan IDs.

Sample Input

The following code sample generates a work plan:

```
{
    "inputs" : [ {
        "recordId" : "0WOxx00000007j3"
    }]
}
```

Sample Output

The following code sample illustrates a response after a successful request.

```
[ {
    "actionName" : "generateWorkPlans",
    "errors" : null,
    "isSuccess" : true,
    "outputValues" : {
        "recordId" : "0WOxx00000007j3",
        "workPlanIdList" : ["0gqxx0000000Adh", "0gqxx0000000Adi"]
    }
} ]
```

Delete Work Plans

URI: `/services/data/vXX.X/actions/standard/deleteWorkPlans`

This action deletes all the work plans (and its child work steps) associated with a WorkOrder or WorkOrderLineItem.

Table 26: Inputs

Input	Details
`recordId`	**Type** string **Description** Required. The ID of the work order or work order line item.

Table 27: Outputs

Output	Details
`recordId`	**Type** string **Description** The ID of the work order or work order line item.
`workPlanIdList`	**Type** array of strings **Description** The ID list for the work plans that were deleted.

Sample Input

The following code deletes a work plan:

```
{
    "inputs" : [ {
        "recordId" : "0WOxxxxxxxxxxxx"
    }]
}
```

Sample Output

The following code sample illustrates a response after a successful request.

```
[ {
    "actionName" : "deleteWorkPlans",
    "errors" : null,
    "isSuccess" : true,
    "outputValues" : {
        "recordId" : "0WORM000000621X",
        "workPlanIdList" : [ "0gqRM0000004CRS" ]
     }
} ]
```