REST API Developer Guide

Version 60.0, Spring '24

@salesforcedocs

Last updated: January 12, 2024

CONTENTS

CHAPTER 1 Introduction to REST API

In this chapter ...

REST API provides you with programmatic access to your data in Salesforce. The flexibility and scalability of REST API make it an excellent choice for integrating Salesforce into your applications and for performing complex operations on a large scale.

Use this guide to set up your deployment environment and learn about advanced details regarding data access. However, understanding and using REST API requires basic familiarity with software development, web services, and the Salesforce user interface.

If you want to get right to the action, the Quick Start guide covers the basics to get you up and running.

If you're looking for more context about Salesforce APIs, check out the list of links.

Tip: Salesforce REST API is designed to work with Salesforce objects. See the Object Reference for the Salesforce Platform for an introduction and more information about Salesforce objects.

SEE ALSO:

Trailhead: Lightning Platform API Basics

Which API Do I Use?

About REST API

REST API is one of several web interfaces that you can use to access your Salesforce data without using the Salesforce user interface. With API access, you can perform operations and integrate Salesforce into your applications as you like.

You can use REST API tools to create, manipulate, and search data in Salesforce by sending HTTP requests to endpoints in Salesforce. Depending on where you send requests, you access and operate on different pieces of information, called resources. Resources include records, query results, metadata, and more.

REST API uses RESTful architecture to provide a straightforward and consistent interface. A primary benefit of REST API is that it doesn't require much tooling to access your data. It's simpler to use than SOAP API but still provides plenty of functionality.

Although REST API is great for accessing and querying records, other Salesforce APIs, such as Bulk 2.0 API, Metadata API, and Connect REST API, offer additional functionality for specific tasks.

SEE ALSO:

REST Resources and Requests

REST API Architecture

Which API Do I Use?

REST API Release Notes

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

For updates and changes that impact the Salesforce Platform, including REST API, see the API Release Notes.

For new, changed, and deprecated calls and other changes specific to REST API, see REST API in the Salesforce Release Notes.

Supported Editions and Required Permissions

To access your Salesforce org and data using the Salesforce API, you need both an org and a user with API access enabled. There's more than one Salesforce Edition that supports API access and multiple ways to grant API permission to a user.

Supported Editions for API Access

API access is enabled by default in Enterprise, Performance, Unlimited, and Developer Edition orgs. Professional Edition orgs can add API access as an add-on. For more information, visit Salesforce Help: Add Products and Licenses with the Your Account App or contact a Salesforce account executive.

If you send an API request to an org without API access, Salesforce returns a `API_DISABLED_FOR_ORG` error.

To protect the configuration and live data in your production org, we recommend using an isolated environment for active development and testing, such as aDeveloper Edition org, sandbox, or scratch org. When ready, you can move successful changes into your production org.

API User Permissions

To make any API call, a user must have the API Enabled permission turned on in the user profile they're assigned. This permission is enabled by default on some profiles, including many profiles available in Developer Edition orgs. In supported editions, you can also use the Salesforce Integration user license to grant system-to-system integration users full org access while limiting them to API-only operations. For more information, see Salesforce Help: Give Integration Users API Only Access

SEE ALSO:

Get your very own Developer Edition

Scratch Orgs

Sandboxes

Salesforce DX Developer Guide

REST Resources and Requests

REST API is based on the usage of resources—pieces of data in Salesforce, such as records, collections of records, query results, metadata, or API information. Each resource is exposed by a uniform resource identifier (URI) and is accessed by sending HTTP requests to the corresponding URI.

Depending on which resource you want to access and how you construct an HTTP request, you can perform several types of operations, including:

- Determine available API versions
- Access limits for your Salesforce org
- Retrieve object metadata
- Create, read, update, and delete records
- Query and search for data

You can send HTTP requests using a variety of software tools, which means that the exact appearance of a request can look different from the cURL examples in this guide. However, no matter how you submit requests, the elements don't change. A typical request consists of these elements.

- URI
- HTTP method
- Headers
- Request body (not required for GET requests)

URIs

The URI is the path to a resource in Salesforce. Although the URI changes from resource to resource, the basic structure remains the same.

```
https://MyDomainName.my.salesforce.com/services/data/vXX.X/resource/
```

Use `https://` to securely access resources.

Replace ***`MyDomainName`*** with the subdomain of your Salesforce org. Salesforce runs on multiple server instances, so the examples in this guide use ***`MyDomainName`*** in place of a specific instance.

Replace ***`XX.X`*** with the version of the API that you want to use. You can find a list of available versions by accessing the Versions on page 134 resource.

Replace ***`resource`*** with the rest of the path to the resource. Depending on the resource, the path can contain parameters, such as IDs to identify a specific record. You can find the URIs for different resources in the Reference section of this guide.

sObject resources access standard and custom objects in Salesforce. For information about objects, see Object Reference for the Salesforce Platform.

Note: Some parts of URIs are case-sensitive, such as IDs. Other parts of URIs aren't case-sensitive, such as object and field names. If your requests aren't successful, check that your URI has the right letter cases by comparing the URI to the examples in this guide.

HTTP Methods

REST API supports standard HTTP request methods (HEAD, GET, POST, PATCH, PUT, and DELETE), which follow the specifications at https://www.w3.org/Protocols/rfc2616/rfc2616-sec9.html.

The purpose of each method varies depending on the resource. For information on how and when to use each method, check the page for that resource in the Reference section of this guide.

Headers

Use headers to pass parameters and customize options for HTTP requests. REST API supports several standard HTTP headers, as well as custom headers that are specific to Salesforce.

Common headers used in the examples include:

- HTTP Accept—Indicates the format that your client accepts for the response body. Possible values are `application/json` and `application/xml`. If the value is missing or malformed, then `application/json` is used by default.
- HTTP Content-type—Indicates the format of the request body that you attach to the request. Possible values are `application/json` and `application/xml`.
- HTTP Authorization—Provides the OAuth 2.0 access token to authorize your client. REST API supports the Bearer authentication type.
- Compression header—Compresses the request or the response. For more information, see Compression Headers on page 9.
- Conditional request header—Validates the records that you access against a precondition. For more information, see Conditional Request Headers on page 10.

Request Bodies

A request body is a rich input that can be included in the request to provide additional information, such as field values for creating or updating records. A request body can be in JSON or XML format.

Note: Resources accessed with the GET method don't require you to attach a request body.

Use the HTTP Content-type header to indicate the file format of the request body. If you use cURL to send the request, attach the JSON or XML file to the request using the `—data-binary` or `-d` option.

SEE ALSO:

Send REST Requests with cURL

Setting Up Your Java Developer Environment

REST API Architecture

REST API follows the standard RESTful principles and characteristics.

Client-server
Client applications are independent from Salesforce REST API, meaning each is managed and updated independently.

Stateless
Each request from client to server must contain all the information necessary to understand the request, and not use any stored context on the server. However, the representations of the resources are interconnected using URIs, which allow the client to progress between states.

Caching behavior
Responses are labeled as cacheable or non-cacheable.

Uniform interface
All resources are accessed with a generic interface over HTTPS.

Named resources
All resources are named using a base URI that follows your Lightning Platform endpoint. See REST Resources and Requests for details and examples.

Layered components
Intermediaries, such as proxy servers and gateways, are allowed between the client and the resources.

In addition to the standard RESTful principles, REST API includes other key characteristics in its architecture that are important to understand and consider as you develop your applications.

Authentication
REST API supports OAuth 2.0 (an open protocol to allow secure API authorization). See Authorize Apps with OAuth in *Salesforce Help* for more details.

Support for JSON and XML
JSON requests are supported in UTF-8 and are the default. XML requests are supported in UTF-8 and UTF-16. XML responses are provided in UTF-8. Use the `HTTP ACCEPT` header to specify either JSON or XML.

In versions 57.0 and earlier, it's possible to append `json` or `xml` to the URI. For example, `/Account/001D000000INjVe.json`. We recommend using the `HTTP ACCEPT` header to specify JSON or XML instead.

In versions 58.0 and later, appending JSON or XML to the URI isn't supported.

Compression
Compression reduces bandwidth loads by compressing the messages sent between REST API and your client. REST API supports compression with gzip and deflate, as defined by the HTTP 1.1 specification. See Compression Headers.

Conditional Requests
Response caching is supported by conditional request headers that follow the standards defined by the HTTP 1.1 specification, with a few exceptions. See Conditional Request Headers.

Cross-Origin Resource Sharing
Cross-Origin Resource Sharing (CORS) enables web browsers to request resources from origins other than their own. For example, using CORS, JavaScript code at https://www.example.com could request a resource from https://www.salesforce.com. To access supported Salesforce APIs, Apex REST resources, and Lightning Out from JavaScript code in a web browser, add the origin serving the code to a Salesforce CORS allowlist. See Perform Cross-Origin Requests from Web Browsers.

Salesforce ID Length
Salesforce IDs in response bodies are always 18-character IDs. In request bodies, you can use either 15 character IDs or 18 character IDs.

Method Overriding

To override an HTTP method if you use an HTTP library that doesn't allow overriding or setting an arbitrary HTTP method name, use the request parameter `_HttpMethod`.

```
POST https://instance_name/services/data/v60.0/chatter/
/chatter/users/me/conversations/03MD0000000008KMAQ
?_HttpMethod=PATCH&read=true
```

Note: The `_HttpMethod` parameter is case-sensitive. Use the correct case for all values.

HTTPS

All communication between client and server is over HTTPS.

Authorization Through Connected Apps and OAuth 2.0

For a client application to access REST API resources, it must be authorized as a safe visitor. To implement this authorization, use a connected app and an OAuth 2.0 authorization flow.

Configure a Connected App

A connected app requests access to REST API resources on behalf of the client application. For a connected app to request access, it must be integrated with your org's REST API using the OAuth 2.0 protocol. OAuth 2.0 is an open protocol that authorizes secure data sharing between applications through the exchange of tokens.

For instructions to configure a connected app, see Create a Connected App in *Salesforce Help*. Specifically, follow the steps in Enable OAuth Settings for API Integration.

Apply an OAuth Authorization Flow

OAuth authorization flows grant a client app restricted access to REST API resources on a resource server. Each OAuth flow offers a different process for approving access to a client app, but in general the flows consist of three main steps.

1. To initiate an authorization flow, a connected app on behalf of a client app requests access to a REST API resource.
2. In response, an authorizing server grants access tokens to the connected app.
3. A resource server validates these access tokens and approves access to the protected REST API resource.

After reviewing and selecting an OAuth authorization flow, apply it to your connected app. For details about each supported flow, see OAuth Authorization Flows in *Salesforce Help*.

More Resources

Salesforce offers the following resources to help you navigate connected apps and OAuth:

- *Salesforce Help*: Connected Apps
- *Salesforce Help*: Authorize Apps with OAuth
- *Salesforce Help*: OpenID Connect Token Introspection
- *Trailhead*: Build Integrations Using Connected Apps

Headers

REST API supports several standard and custom HTTP headers, including both request headers and response headers.

IN THIS SECTION:

Assignment Rule Header

The Assignment Rule header is a request header applied when creating or updating Accounts, Cases, or Leads. If enabled, the active assignment rules are used. If disabled, the active assignment rules are not applied. If a valid AssignmentRule ID is provided, the AssignmentRule is applied. If the header is not provided with a request, REST API defaults to using the active assignment rules.

Call Options Header

Specifies options for the client you're using to access REST API resources. For example, you can provide a default namespace prefix so that you don't need to specify the prefix in your code.

Compression Headers

Use a compression header to compress a REST API request or response. Compression reduces the bandwidth required for a request, although it requires more processing power at your client. In most cases, this tradeoff benefits the overall performance of your application.

Conditional Request Headers

Use a conditional request header to validate resources before accessing them. By setting a precondition in the header, you ensure that your request succeeds only if that precondition is met. This functionality helps you prevent mistakes and reject outdated requests when updating Salesforce data. You can implement a variety of techniques with conditional request headers, such as response caching.

Duplicate Rule Header

Configure options for duplicate rules. Salesforce uses duplicate rules to see if the record that is being created, updated, or upserted is a duplicate of an existing record. Duplicate rules are part of Duplicate Management.

Limit Info Header

This response header is returned in each request to REST API (except for calls to the Versions URI, `/`, which do not count towards your org's limit). You can use the information to monitor API limits.

MRU Header

Defines if the record that is created, updated, upserted, or retrieved updates the list of most recently used (MRU) items. MRU items are shown in the Recent Items section of the sidebar in the Salesforce user interface. This header is available in API version 60.0 and later.

Package Version Header

Specifies the version of each package referenced by a client. A package version is a number that identifies the set of components and behavior contained in a package. This header can also be used to specify a package version when making calls to an Apex REST web service.

Query Options Header

Specifies options used in a query, such as the query results batch size. Use this request header with the Query resource.

Warning Header

This header is returned if there are warnings, such as the use of a deprecated version of the API.

Assignment Rule Header

The Assignment Rule header is a request header applied when creating or updating Accounts, Cases, or Leads. If enabled, the active assignment rules are used. If disabled, the active assignment rules are not applied. If a valid AssignmentRule ID is provided, the AssignmentRule is applied. If the header is not provided with a request, REST API defaults to using the active assignment rules.

Note: This header also gets applied when making REST API calls that indirectly result in creating or updating Accounts, Cases, or Leads. For example, if you use this header with a call that updates a record, and the update executes an Apex trigger that updates a Case, the assignment rules would be applied.

Header Field Name and Values

Field name

`Sforce-Auto-Assign`

Field values

- `TRUE`. Active assignment rules are applied for created or updated Accounts, Cases, or Leads.
- `FALSE`. Active assignment rules are not applied for created or updated Accounts, Cases, or Leads.
- Valid AssignmentRule ID. The given AssignmentRule is applied for created Accounts, Cases, or Leads.

`TRUE` and `FALSE` are not case-sensitive.

If the header is not provided in the request, the default value is `TRUE`.

Example

`Sforce-Auto-Assign: FALSE`

Call Options Header

Specifies options for the client you're using to access REST API resources. For example, you can provide a default namespace prefix so that you don't need to specify the prefix in your code.

The Call Options header can be used with sObject Basic Information, sObject Rows, sObject Rows by External ID, Query, QueryAll, and Search. It is also supported in Bulk API and Bulk API 2.0.

Header Field Name and Values

Field name

`Sforce-Call-Options`

Field values

- `client`—A string used as an identifier for the client sending the request. This string appears in log files, helping you keep track of which client sent a request.
- `defaultNamespace`—A developer namespace prefix used as the default namespace for the request. With this header field, the request resolves field names in managed packages without specified namespaces. (Not supported in Bulk API.)

Example

If the developer namespace prefix is `battle`, and you have a custom field called `botId` in a package, set the default namespace with the call options header:

```
Sforce-Call-Options: client=caseSensitiveToken; defaultNamespace=battle
```

Then queries such as the following succeed:

```
/services/data/vXX.X/query/?q=SELECT+Id+botID__c+FROM+Account
```

In this case, the actual field queried is the `battle__botId__c` field.

Using this header allows you to write client code without having to specify the namespace prefix. In the previous example, without the header you must write `battle__botId__c`.

If this field is set, and the query also specifies the namespace, the response doesn't include the prefix. For example, if you set this header to `battle`, and issue a query like `SELECT+Id+battle__botID__c+FROM+Account`, the response uses a `botId__c` element, not a `battle_botId__c` element.

The `defaultNamespace` field is ignored when retrieving describe information, which avoids ambiguity between namespace prefixes and customer fields of the same name.

Compression Headers

Use a compression header to compress a REST API request or response. Compression reduces the bandwidth required for a request, although it requires more processing power at your client. In most cases, this tradeoff benefits the overall performance of your application.

REST API supports the gzip and deflate compression algorithms, as defined by the HTTP 1.1 specification. If you're unsure about which one to use, gzip is more common than deflate.

Tip: For better performance, we suggest that clients accept and support compression as defined by the HTTP 1.1 specification.

Request Compression

Include a `Content-Encoding: gzip` or `Content-Encoding: deflate` header to compress a request. REST API decompresses any requests before processing.

This example request is compressed with gzip.

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/ -H
"Authorization: Bearer access-token" -H "Content-Type: application/json" -H
"Content-Encoding: gzip" —data-binary @new-account.json -X POST
```

Response Compression

Salesforce compresses a response only if the request contains an `Accept-Encoding: gzip` or `Accept-Encoding: deflate` header. REST API isn't required to compress the response even if you've specified Accept-Encoding, but it normally does. If compressed, the response contains a Content-Encoding header with the compression algorithm so that your client knows to decompress it.

This example request asks for a compressed response.

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/0015e000009sS0DAAU
 -H "Authorization: Bearer access-token" -H "Content-Type: application/json" -H
"Accept-Encoding: gzip" -X GET
```

Conditional Request Headers

Use a conditional request header to validate resources before accessing them. By setting a precondition in the header, you ensure that your request succeeds only if that precondition is met. This functionality helps you prevent mistakes and reject outdated requests when updating Salesforce data. You can implement a variety of techniques with conditional request headers, such as response caching.

Two types of validation are available with conditional request headers: strong and weak. Strong validation checks whether the precondition exactly matches the resource in Salesforce. Strong validation headers include `If-Match` and `If-None-Match`, which use entity tags (ETags) to compare the precondition to the record in Salesforce.

Weak validation checks a precondition against the resource in Salesforce, but it doesn't guarantee that the two are identical. Weak validation headers include `If-Modified-Since` or `If-Unmodified-Since`, which compare the precondition to the last modified date of the record in Salesforce.

REST API conditional headers follow the HTTP 1.1 specification with the following exceptions.

- When you include an invalid header value for `If-Match`, `If-None-Match`, or `If-Unmodified-Since` on a PATCH or POST request, a `400 Bad Request` status code is returned.
- The `If-Range` header isn't supported.
- DELETE requests aren't supported

ETag

The `ETag` header is a response header that's returned when you access the sObject Rows resource. It's a hash of the content that's used by the `If-Match` and `If-None-Match` request headers in subsequent requests to determine if the content has changed.

This header is supported by sObject Rows (Account records only) resources.

This example shows an `ETag` returned by REST API.

```
ETag: "U5iWijwWbQD18jeiXwsqxeGpZQk=-gzip"
```

You can find the HTTP 1.1 specification for the `ETag` header at www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.19 .

If-Match

The `If-Match` header is a request header for sObject Rows that includes a list of ETags. If the ETag of the record that you're requesting matches an ETag specified as a precondition in the header, the request is processed. Otherwise, a 412 Precondition Failed status code is returned, and the request isn't processed.

This header supports sObject Rows (Account records only) resources.

In this example an, `If-Match` header is included with a request.

```
If-Match: "Jbjuzw7dbhaEG3fd90kJbx6A0ow=-gzip", "U5iWijwWbQD18jeiXwsqxeGpZQk=-gzip"
```

You can find the HTTP 1.1 specification for the `If-Match` header at www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.24 .

If-None-Match

The `If-None-Match` header is a request header for sObject Rows that's the inverse of `If-Match`. If the ETag of the record that you're requesting matches an ETag specified in the header, the request isn't processed. A 304 Not Modified status code is returned for GET or HEAD requests, and a 412 Precondition Failed status code is returned for PATCH requests.

This header supports sObject Rows (Account records only) resources.

In this example, an `If-None-Match` header is included with a request.

```
If-None-Match: "Jbjuzw7dbhaEG3fd90kJbx6A0ow=-gzip", "U5iWijwWbQD18jeiXwsqxeGpZQk=-gzip"
```

You can find the HTTP 1.1 specification for the `If-None-Match` header at www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.26 .

If-Modified-Since

The `If-Modified-Since` header is a time-based request header. The request is processed only if the data has changed since the date and time specified in the header. Otherwise, a 304 Not Modified status code is returned, and the request isn't processed.

This header supports sObject Rows, sObject Describe, Describe Global, and Invocable Actions resources.

In this example an `If-Modified-Since` header is included with a request.

```
If-Modified-Since: Tue, 10 Aug 2015 00:00:00 GMT
```

You can find the HTTP 1.1 specification for the `If-Modified-Since` header at www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.25 .

If-Unmodified-Since

The `If-Unmodified-Since` header is a request header that's the inverse of `If-Modified-Since`. If you make a request and include the If-Unmodified-Since header, REST API processes the request only if the data hasn't changed since the specified date. Otherwise, a 412 Precondition Failed status code is returned, and the request isn't processed.

This header supports sObject Rows, sObject Describe, Describe Global, and Invocable Actions resources.

In this example, an `If-Unmodified-Since` header is included in a request.

```
If-Unmodified-Since: Tue, 10 Aug 2015 00:00:00 GMT
```

You can find the HTTP 1.1 specification for the `If-Unmodified-Since` header at www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.28 .

Duplicate Rule Header

Configure options for duplicate rules. Salesforce uses duplicate rules to see if the record that is being created, updated, or upserted is a duplicate of an existing record. Duplicate rules are part of Duplicate Management.

This header is available in API version 52.0 and later.

Header Field Name and Values

The default value for all fields is `false`.

Field name

`allowSave`

Field values

- `true`—allow the user to acknowledge the alert and save the duplicate record. Applicable if an alert is enabled for the action.
- `false`—don't allow the user to acknowledge the alert or save the duplicate record. Applicable if an alert is enabled for the action.

Field name

`includeRecordDetails`

Field values

- `true`—return all fields in the duplicate record.
- `false`—return the duplicate record ID, but not the fields.

Field name

`runAsCurrentUser`

Field values

- `true`—when the duplicate rule is run, use the current user's sharing rules.
- `false`—when the duplicate rule is run, use the system sharing rules, not the current user's sharing rules.

Example

Allow the user to acknowledge the alert and save the duplicate record. Indicate that the duplicate field's records are returned, and that the current user's sharing rules are enforced. These duplicate management configuration options are now applied when records are created, updated, and upserted.

```
Sforce-Duplicate-Rule-Header: allowSave=true, includeRecordDetails=true,
runAsCurrentUser=true
```

Limit Info Header

This response header is returned in each request to REST API (except for calls to the Versions URI, `/`, which do not count towards your org's limit). You can use the information to monitor API limits.

Header Field Name and Values

Field name

`Sforce-Limit-Info`

Field values

- `api-usage`—Specifies the daily API usage for the organization against which the call was made. The first number is the number of API calls used, and the second number is the API limit for the organization.

The values returned in the header represent standard REST API limits and usage, except when calls are made using Salesforce Functions. Calls made using Salesforce Functions draw from a Functions-specific allocation.

Example

`Sforce-Limit-Info: api-usage=10018/100000`

SEE ALSO:

Salesforce Functions Guide: Functions Limits

MRU Header

Defines if the record that is created, updated, upserted, or retrieved updates the list of most recently used (MRU) items. MRU items are shown in the Recent Items section of the sidebar in the Salesforce user interface. This header is available in API version 60.0 and later.

Note: REST APIs that create, update, upsert, or retrieve records support this header. For examples of these APIs, see sObject Rows and sObject Rows by External ID in the See Also section. REST APIs used to query records via SOQL never update the MRU, so the header field has no impact on the MRU and is always set to `false`.

Header Field Name and Values

The default value for all fields is `true`, except for with REST APIs that query records via SOQL. The default value is subject to change.

For integration users who make only API calls, Salesforce recommends setting this field to `false` to improve performance. MRU isn't relevant for integration users using solely APIs because they don't use the Salesforce user interface.

Field name

`updateMru`

Field values

- `true`—Update MRU with related records if the No MRU Updates field isn't enabled for the user on the Users page in Setup. For more information on how to enable this field, see the Edit Users page in the See Also section.
- `false`—Don't update MRU with related records.

Example

Indicates that the record updates the MRU.

```
Sforce-Mru: updateMru=true
```

SEE ALSO:

sObject Rows

sObject Rows by External ID

Salesforce Help: Edit Users

Package Version Header

Specifies the version of each package referenced by a client. A package version is a number that identifies the set of components and behavior contained in a package. This header can also be used to specify a package version when making calls to an Apex REST web service.

The Package Version header can be used with the following resources: Describe Global, sObject Describe, sObject Basic Information, sObject Rows, sObject Layouts, Query, QueryAll, Search, and sObject Rows by External ID.

Header Field Name and Values

Field name and value

`x-sfdc-packageversion-[namespace]: xx.x`, where `[namespace]` is the unique namespace of the managed package and `xx.x` is the package version.

Example

`x-sfdc-packageversion-clientPackage: 1.0`

Query Options Header

Specifies options used in a query, such as the query results batch size. Use this request header with the Query resource.

Header Field Name and Values

Field name

`Sforce-Query-Options`

Field values

- `batchSize`—A numeric value that specifies the number of records returned for a query request. Child objects count toward the number of records for the batch size. For example, in relationship queries, multiple child objects are returned per parent row returned.

  The default is 2,000; the minimum is 200, and the maximum is 2,000. There is no guarantee that the requested batch size is the actual batch size. Changes are made as necessary to maximize performance.

Example

`Sforce-Query-Options: batchSize=1000`

Warning Header

This header is returned if there are warnings, such as the use of a deprecated version of the API.

Header Field Name and Values

Field name

`Warning`

Field Values

`warningCode warningMessage`

For warnings about deprecated API versions, the *`warningCode`* is `299`.

Example

```
Warning: 299 - "This API is deprecated and will be removed by Summer '22. Please see
https://help.salesforce.com/articleView?id=000351312 for details."
```

Send REST Requests with cURL

The examples in this guide use the cURL tool to send HTTP requests that access, create, and manipulate resources in Salesforce. If you use a different tool to send requests, you can use the same elements from the cURL examples to send requests.

cURL is pre-installed on many Linux and macOS systems. Windows users can download a version at curl.haxx.se/. When using HTTPS on Windows, ensure that your system meets the cURL requirements for SSL.

Note: cURL is an open-source tool and isn't supported by Salesforce.

Attaching Request Bodies

Many examples include request bodies—JSON or XML files that contain data for the request. When using cURL, save these files to your local system and attach them to the request using the —data-binary or -d option.

This example attaches the `new-account.json` file.

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/ -H
"Authorization Bearer access-token" -H “Content-Type: application/json” —data-binary
@new-account.json -X POST
```

Handling Exclamation Marks in Access Tokens

When you run cURL examples, you can get an error on macOS and Linux systems due to the presence of the exclamation mark (!) special character in OAuth access tokens. To avoid getting this error, either escape the exclamation mark or use single quotes.

To escape the exclamation mark in the access token, insert a backslash before it (\!) when the access token is enclosed within double quotes. For example, the access token string in this cURL command has the exclamation mark (!) escaped.

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/ -H "Authorization: Bearer

00D50000000IehZ\!AQcAQH0dMHZfz972Szmpkb58urFRkgeBGsxL_QJWwYMfAbUeeG7c1E6LYUfiDUkWe6H34r1AAwOR8B8fLEz6n04NPGRrq0FM"
```

Or, you can enclose the access token within single quotes.

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/ -H 'Authorization: Bearer

00D50000000IehZ!AQcAQH0dMHZfz972Szmpkb58urFRkgeBGsxL_QJWwYMfAbUeeG7c1E6LYUfiDUkWe6H34r1AAwOR8B8fLEz6n04NPGRrq0FM'
```

SEE ALSO:

Setting Up Your Java Developer Environment

Configure Salesforce CORS Allowlist

EDITIONS

Available in: Salesforce Classic (not available in all orgs) and Lightning Experience

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

Available with API access enabled in: Professional Edition

USER PERMISSIONS

To create, read, update, and delete:

- Modify All Data

Cross-Origin Resource Sharing (CORS) allows web browsers to request resources from other origins. For example, using CORS, the JavaScript for a web application at `https://www.example.com` can request a resource from `https://www.salesforce.com`. To allow access to supported Salesforce APIs, Apex REST resources, and Lightning Out from JavaScript code in a web browser, add the requesting origin to your Salesforce CORS allowlist. For Lightning apps that allow web browsers to make requests from their orgs, CORS allowlist prevents requests to Lightning apps unless the request comes from an approved URL.

These Salesforce technologies support CORS.

- Apex REST
- Bulk API
- Bulk API 2.0
- Connect REST API
- Lightning Out
- REST API
- CRM Analytics REST API
- User Interface API

Add an origin serving the request code to the CORS allowlist. If a browser that supports CORS makes a request to an origin in the allowlist, Salesforce returns the origin in the

`Access-Control-Allow-Origin` HTTP header along with any additional CORS HTTP headers. If the origin isn't included in the allowlist, Salesforce returns HTTP status code 403.

1. From Setup, in the Quick Find box, enter *CORS*, and then select **CORS**.
2. Select **New**.
3. Enter a resource in Origin URL Pattern.

   Tip: The origin URL pattern doesn't always match the URL that appears in your browser's address bar.

4. Save your changes.

The origin URL pattern must include the HTTPS protocol (unless you're using your localhost) and a domain name. It can also include a port. The wildcard character (*) is supported and must be in front of a second-level domain name. For example, `https://*.example.com` adds all subdomains of `example.com` to the allowlist.

The origin URL pattern can be an IP address. But an IP address and a domain that resolve to the same address aren't the same origin, and you must add them to the CORS allowlist as separate entries.

Google Chrome™ and Mozilla® Firefox® browser extensions are allowed as resources in API version 53 (Winter '22) or later . Chrome extensions must use the prefix `chrome-extension://` and 32 characters without digits or capital letters, for example `chrome-extension://abdkkegmcbiomijcbdaodaflgehfffed`. Firefox extensions must use the prefix `moz-extension://` and an 8-4-4-4-12 format of small alphanumeric characters, for example `moz-extension://1234ab56-78c9-1df2-3efg-4567891hi1j2`.

You can get a successful response when requesting a REST resource in a CORS preflight test, but receive an unsuccessful response to the actual request. This discrepancy can occur when the resource is deleted after the preflight test and before the request is made. It can also occur if the resource doesn't exist. A CORS preflight confirms if resources can be passed between servers, but doesn't check if a specific resource exists or not. CORS preflight requests are typically issued automatically by a browser.

Note: To access certain OAuth endpoints with CORS, other requirements apply. See https://help.salesforce.com/s/articleView?id=sf.remoteaccess_oauth_endpoints_cors.htm.

Valid Date and DateTime Formats

Specify the right format for `dateTime` and `date` fields.

dateTime

Use the *yyyy-MM-ddTHH:mm:ss.SSS+/-HH:mm* or *yyyy-MM-ddTHH:mm:ss.SSSZ* formats to specify `dateTime` fields.

- `yyyy` is the four-digit year
- `MM` is the two-digit month (01-12)
- `dd` is the two-digit day (01-31)
- 'T' is a separator indicating that time-of-day follows
- `HH` is the two-digit hour (00-23)
- `mm` is the two-digit minute (00-59)
- `ss` is the two-digit seconds (00-59)
- `SSS` is the optional three-digit milliseconds (000-999)
- `+/-HH:mm` is the Zulu (UTC) time zone offset

- 'Z' is the reference UTC timezone

When a timezone is added to a UTC `dateTime`, the result is the date and time in that timezone. For example, 2002-10-10T12:00:00+05:00 is 2002-10-10T07:00:00Z and 2002-10-10T00:00:00+05:00 is 2002-10-09T19:00:00Z. See W3C XML Schema Part 2: DateTime Datatype.

date

Use the *`yyyy-MM-dd`* format to specify `date` fields.

Note: Specifying an offset for `date` is not supported.

Status Codes and Error Responses

Either when an error occurs or when a response is successful, the response header contains an HTTP code, and the response body usually contains:

- The HTTP response code
- The message accompanying the HTTP response code
- The field or object where the error occurred (if the response returns information about an error)

HTTP response code	Description
200	"OK" success code, for GET, HEAD, and some PATCH requests.
201	"Created" success code, for POST requests and some PATCH requests.
204	"No Content" success code, for DELETE requests and some PATCH requests.
300	The value returned when an external ID exists in more than one record. The response body contains the list of matching records.
304	The request content hasn't changed since a specified date and time. The date and time is provided in a `If-Modified-Since` header. See Get Object Metatdata Changes for an example.
400	The request couldn't be understood, usually because the JSON or XML body contains an error.
401	The session ID or OAuth token used has expired or is invalid. The response body contains the `message` and `errorCode`.
403	The request has been refused. Verify that the logged-in user has appropriate permissions. If the error code is REQUEST_LIMIT_EXCEEDED, you've exceeded API request limits in your org.
404	The requested resource couldn't be found. Check the URI for errors, and verify that there are no sharing issues.
405	The method specified in the Request-Line isn't allowed for the resource specified in the URI.
409	The request couldn't be completed due to a conflict with the current state of the resource. Check that the API version is compatible with the resource you're requesting.
410	The requested resource has been retired or removed. Delete or update any references to the resource.

HTTP response code	Description
412	The request wasn't executed because one or more of the preconditions that the client specified in the request headers wasn't satisfied. For example, the request includes an `If-Unmodified-Since` header, but the data was modified after the specified date.
414	The length of the URI exceeds the 16,384-byte limit.
415	The entity in the request is in a format that's not supported by the specified method.
420	Salesforce Edge doesn't have routing information available for this request host. Contact Salesforce Customer Support.
428	The request wasn't executed because it wasn't conditional. Add one of the Conditional Request Headers, such as `If-Match`, to the request and resubmit it.
431	The combined length of the URI and headers exceeds the 16,384-byte limit.
500	An error has occurred within Lightning Platform, so the request couldn't be completed. Contact Salesforce Customer Support.
502	Salesforce Edge wasn't able to communicate successfully with the Salesforce instance.
503	The server is unavailable to handle the request. Typically this issue occurs if the server is down for maintenance or is overloaded.

Incorrect ID example

Using a non-existent ID in a request using JSON or XML (*request_body*.json or *request_body*.xml)

```
[
{
  "fields" : [ "Id" ],
  "message" : "Account ID: id value of incorrect type: 001900K0001pPuOAAU",
  "errorCode" : "MALFORMED_ID"
}
]
```

Resource doesn't exist

Requesting a resource that doesn't exist, for example, you try to create a record using a misspelled object name

```
[
{
  "message" : "The requested resource does not exist",
  "errorCode" : "NOT_FOUND"
}
]
```

API End-of-Life Policy

See which REST API versions are supported, unsupported, or unavailable.

Salesforce is committed to supporting each API version for a minimum of 3 years from the date of first release. To improve the quality and performance of the API, versions that are over 3 years old sometimes are no longer supported.

Salesforce notifies customers who use an API version scheduled for deprecation at least 1 year before support for the version ends.

Salesforce API Versions	Version Support Status	Version Retirement Info
Versions 31.0 through 60.0	Supported.	
Versions 21.0 through 30.0	As of Summer '22, these versions have been deprecated and no longer supported by Salesforce. Starting Summer '25, these versions will be retired and unavailable.	Salesforce Platform API Versions 21.0 through 30.0 Retirement
Versions 7.0 through 20.0	As of Summer '22, these versions are retired and unavailable.	Salesforce Platform API Versions 7.0 through 20.0 Retirement

If you request any resource or use an operation from a retired API version, REST API returns the `410:GONE` error code.

To identify requests made from old or unsupported API versions, use the API Total Usage event type.

CHAPTER 2 Quick Start

In this chapter ...

To set up and run REST API, send a few basic requests to Salesforce. This Quick Start explains setting up a basic environment and updating a record using REST API. You can set up and use REST API in many ways, and the examples show how to use the free Developer Edition and cURL.

Using cURL

Get to know the formatting that you can use with cURL to make requests to Salesforce. This Quick Start uses cURL examples, but you can use any tool or development environment that can make REST requests.

Familiarize yourself with cURL enough to be able to understand the examples in this guide and translate them into the tool that you're using. To attach files containing the body of the request, you must properly format the access token. Use these tips to help you use cURL while working through the Quick Start. For more information about cURL, see curl.se.

Attach Request Bodies

Many examples include request bodies—JSON or XML files that contain data for the request. When using cURL, save these files to your local system and attach them to the request using the —`data-binary` or `-d` option.

This example attaches the `new-account.json` file.

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/ -H
'Authorization Bearer
00DE0X0A0M0PeLE!AQcAQH0dMHEXAMPLEzmpkb58urFRkgeBGsxL_QJWwYMfAbUeeG7c1EXAMPLEDUkWe6H34r1AAwOR8B8fLEz6nEXAMPLE'
 -H "Content-Type: application/json" —d @new-account.json -X POST
```

Handle Exclamation Marks in Access Tokens

When you run cURL examples, you can get an error on macOS and Linux systems due to the presence of the exclamation mark (!) in OAuth access tokens. To avoid getting this error, either escape the exclamation mark or use single quotes. To escape the exclamation mark in the access token, insert a backslash before it when the access token is enclosed in double quotes.

```
\!
```

For example, the access token string in this cURL command has the exclamation mark (!) escaped.

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/ -H "Authorization: Bearer

00DE0X0A0M0PeLE\!AQcAQH0dMHEXAMPLEzmpkb58urFRkgeBGsxL_QJWwYMfAbUeeG7c1EXAMPLEDUkWe6H34r1AAwOR8B8fLEz6nEXAMPLE"
```

Or, you can enclose the access token within single quotes to not escape the exclamation mark.

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/ -H 'Authorization: Bearer

00DE0X0A0M0PeLE!AQcAQH0dMHEXAMPLEzmpkb58urFRkgeBGsxL_QJWwYMfAbUeeG7c1EXAMPLEDUkWe6H34r1AAwOR8B8fLEz6nEXAMPLE'
```

Important: All quotes, whether single or double, must be straight quotes, not curly quotes.

Step One: Sign up for Salesforce Developer Edition

Developer Edition provides a free and easy solution so that you can use Salesforce for testing and development.

To sign up for a Developer Edition account, go to `developer.salesforce.com/signup`.

Note: The Developer Edition data storage maximum is 5 MB. This limit doesn't prevent you from working with these examples.

If you have a development sandbox, you can use it with these examples.

Before you begin, verify that your user profile has the API Enabled permission by following the instructions in User Permissions in *Salesforce Help*.

SEE ALSO:

User Permissions

Knowledge Article: Salesforce editions with API access

Step Two: Set Up Authentication

To successfully send requests, REST API requires an access token obtained by authentication. Although you can create and authenticate against your own connected app, these Quick Start examples use Salesforce CLI for convenience. Salesforce CLI is a connected app that you can authenticate, and it requires no work to configure.

Get an Access Token with Salesforce CLI

Use the access token (also known as a "bearer token") that you get from Salesforce CLI to authenticate cURL requests.

1. Install or update Salesforce CLI. .
   a. If you already have Salesforce CLI installed, update it using the instructions in Update Salesforce CLI.
   b. If you need to Install Salesforce CLI, install the latest version for your operating system.
   c. Verify Your Installation.
2. Log in to your Developer org with Salesforce CLI.

```
sf org login web
```

   A browser opens to https://login.salesforce.com.
3. In the browser, log in to your Developer org with your user's credentials.
4. In the browser, click **Allow** to allow access.

   At the command line, you see a similar confirmation message.

```
Successfully authorized juliet.capulet@empathetic-wolf-g5qddtr.com with org ID
00D5fORGIDEXAMPLE
```

5. At the command line, get the access token by viewing authentication information about your org.

```
sf org display --target-org <username>
```

   For example:

```
sf org display --target-org juliet.capulet@empathetic-wolf-g5qddtr.com
```

   Example command output:

```
=== Org Description

 KEY              VALUE

 ───────────────
────────────────────────────────────────────────────────────────────────────────────────
```

```
 Access Token
00DE0X0A0M0PeLE!AQcAQH0dMHEXAMPLEzmpkb58urFRkgeBGsxL_QJWwYMfAbUeeG7c1EXAMPLEDUkWe6H34r1AAwOR8B8fLEz6nEXAMPLE

 Api Version      59.0
 Client Id        PlatformCLI
 Created By       jules@sf.com
 Created Date     2023-11-16T20:35:21.000+0000
 Dev Hub Id       jules@sf.com
 Edition          Developer
 Expiration Date  2023-11-23
 Id               00D5fORGIDEXAMPLE
 Instance Url     https://MyDomainName.my.salesforce.com
 Org Name         Dreamhouse
 Signup Username  juliet.capulet@empathetic-wolf-g5qddtr.com
 Status           Active
 Username         juliet.capulet@empathetic-wolf-g5qddtr.com
```

In the command output, make note of the long Access Token string and the Instance Url string. You need both to make cURL requests.

Note: To get a new token after your access token expires, repeat this step of viewing your authentication information.

Optional Salesforce CLI Shortcuts

After you've authenticated successfully, try out these optional shortcuts in your cURL workflow to streamline future authentication with the Salesforce CLI.

List My Orgs

```
sf org list
```

Lists all the orgs that you've created or authenticated to.

Open My Org

```
sf org open --target-org <username>
```

Opens the specified org (identified by username or alias) in your browser. Because you've successfully authenticated with this org previously using the `org login web` Salesforce CLI command, it's not required to provide your credentials again.

Display the Access Token for My Org

```
sf org display --target-org <username>
```

Output includes your access token, client ID, connected status, org ID, instance URL, username, and alias, if applicable.

Set an Alias for My Username

For convenience, create an alias for your username so that you don't have to enter the entire Salesforce string. For example, instead of

```
juliet.capulet@empathetic-wolf-g5qddtr.com
```

Create an alias like

```
dev
```

To set the alias in this example, run

```
sf alias set dev juliet.capulet@empathetic-wolf-g5qddtr.com
```

Use These Commands in a Script

Use the CLI's JSON output by invoking the `--json` flag. Requesting JSON output provides a consistent output format, which is ideal for running scripts. Without the `--json` flag, the CLI can change the output format.

See Also

- Salesforce CLI Setup Guide
- Salesforce CLI Command Reference

SEE ALSO:

Salesforce Help: Authorize Apps with OAuth

Step Three: Walk Through the Sample Code

To access different types of resources in Salesforce, make a series of REST requests. Before you try these examples, make sure to complete the prerequisites and obtain an access token in Step 1 of this Quick Start.

You can copy and paste these examples to send them with cURL. But first replace *`MyDomainName`* in the base URI with your Salesforce domain. For information on the anatomy of a REST request, see REST Resources and Requests on page 3.

Get the Salesforce Version

To retrieve information about each available Salesforce version, submit a Versions request. In this case, the request doesn't require authentication.

```
curl https://MyDomainName.my.salesforce.com/services/data/
```

The output from this request, including the response header, specifies all valid versions. Your result can include more than one value.

```
Content-Length: 88
Content-Type: application/json;
charset=UTF-8 Server:
[
    {
        "label":"Spring '11",
        "url":"/services/data/v21.0",
        "version":"21.0"
    }
    ...
]
```

Get a List of Resources

To retrieve a list of the resources available for Salesforce, in this example, for version 60.0, submit a Resources by Version request.

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/ -H "Authorization: Bearer
 access_token"
```

The output from this request shows that `sobjects` is one of the available resources in Salesforce version 60.0.

```
{
    "sobjects" : "/services/data/v60.0/sobjects",
    "search" : "/services/data/v60.0/search",
    "query" : "/services/data/v60.0/query",
    "recent" : "/services/data/v60.0/recent"
}
```

Get a List of Available Objects

To request a list of the available objects, submit a Describe Global request.

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/ -H "Authorization:
 Bearer access_token"
```

The output from this request shows that the Account object is available.

```
Transfer-Encoding: chunked
Content-Type: application/json;
charset=UTF-8 Server:
{
 "encoding" : "UTF-8",
 "maxBatchSize" : 200,
 "sobjects" : [ {
    "name" : "Account",
    "label" : "Account",
    "custom" : false,
    "keyPrefix" : "001",
    "updateable" : true,
    "searchable" : true,
    "labelPlural" : "Accounts",
    "layoutable" : true,
    "activateable" : false,
    "urls" : { "sobject" : "/services/data/v60.0/sobjects/Account",
    "describe" : "/services/data/v60.0/sobjects/Account/describe",
    "rowTemplate" : "/services/data/v60.0/sobjects/Account/{ID}" },
    "createable" : true,
    "customSetting" : false,
    "deletable" : true,
    "deprecatedAndHidden" : false,
    "feedEnabled" : false,
    "mergeable" : true,
    "queryable" : true,
    "replicateable" : true,
    "retrieveable" : true,
    "undeletable" : true,
    "triggerable" : true },
   },
...
```

Get Basic Object Information

To retrieve basic information about the available Account object's metadata, submit a sObject Basic Information request.

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/ -H
"Authorization: Bearer access_token"
```

The output from this request shows basic attributes of the Account object such as its name and label, and it lists the most recently used accounts.


```
{
    "objectDescribe" :
    {
        "name" : "Account",
        "updateable" : true,
        "label" : "Account",
        "keyPrefix" : "001",

        ...

        "replicateable" : true,
        "retrieveable" : true,
        "undeletable" : true,
        "triggerable" : true
    },
    "recentItems" :
    [
        {
            "attributes" :
            {
                "type" : "Account",
                "url" : "/services/data/v60.0/sobjects/Account/001D000000INjVeIAL"
            },
            "Id" : "001D000000INjVeIAL",
            "Name" : "asdasdasd"
        },

        ...

    ]
}
```


Get a List of Fields

To retrieve more detailed information, submit a sObject Describe request.

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/describe/
 -H "Authorization: Bearer access_token"
```

The output from this request shows more detailed information about the Account object, such as its field attributes and child relationships.


```
{
    "name" : "Account",
    "fields" :
    [
```

```
        {
            "length" : 18,
            "name" : "Id",
            "type" : "id",
            "defaultValue" : { "value" : null },
            "updateable" : false,
            "label" : "Account ID",
            ...
        },
        ...
    ],
    "updateable" : true,
    "label" : "Account",
    ...
    "urls" :
    {
        "uiEditTemplate" : "https://MyDomainName.my.salesforce.com/{ID}/e",
        "sobject" : "/services/data/v60.0/sobjects/Account",
        "uiDetailTemplate" : "https://MyDomainName.my.salesforce.com/{ID}",
        "describe" : "/services/data/v60.0/sobjects/Account/describe",
        "rowTemplate" : "/services/data/v60.0/sobjects/Account/{ID}",
        "uiNewRecord" : "https://MyDomainName.my.salesforce.com/001/e"
    },
    "childRelationships" :
    [
        {
            "field" : "ParentId",
            "deprecatedAndHidden" : false,
            ...
        },
        ...
    ],

    "createable" : true,
    "customSetting" : false,
    ...
}
```

Execute a SOQL Query

To execute a SOQL query that retrieves a list of all the Account name values, submit a Query request.

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/query?q=SELECT+name+from+Account
 -H "Authorization: Bearer access_token"
```

The output lists the available Account names, and each name's preceding attributes include the Account IDs.

```
{
    "done" : true,
    "totalSize" : 14,
    "records" :
    [
        {
```

```
            "attributes" :
            {
                "type" : "Account",
                "url" : "/services/data/v60.0/sobjects/Account/001D000000IRFmaIAH"
            },
            "Name" : "Test 1"
        },
        {
            "attributes" :
            {
                "type" : "Account",
                "url" : "/services/data/v60.0/sobjects/Account/001D000000IomazIAB"
            },
            "Name" : "Test 2"
        },
        ...
    ]
}
```

Note: You can find more information about SOQL in the *Salesforce SOQL and SOSL Reference Guide*.

Update a Field on a Record

To retrieve one of the accounts and update its billing city, submit an sObject Rows request. To update the object, create a text file called `patchaccount.json` containing the new billing city information.

```
{
    "BillingCity" : "Fremont"
}
```

Specify this JSON file in the REST request. The cURL notation requires the `—d` option when specifying data. For more information, see *http://curl.haxx.se/docs/manpage.html*.

Also, specify the `PATCH` method, which is used to update a REST resource. This cURL command retrieves the specified Account object using its ID field and updates its billing city.

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/001D000000IroHJ
 -H "Authorization: Bearer access_token" -H "Content-Type: application/json" --data-binary
 @patchaccount.json -X PATCH
```

No response body is returned, just the headers.

```
HTTP/1.1 204 No Content
Server:
Content-Length: 0
```

To see that the billing address has changed to Fremont, refresh the page on the account.

Using Other Tools

Other tools are available to obtain data from your Salesforce org.

If you don't want to use CURL, you can use other tools to exercise the API. Possible choices include:

- Salesforce CLI.
- Postman (a third-party tool). For more information, see the Postman API Client Trailhead module.

These tools provide functionality that simplifies the process of creating and sending many API requests.

CHAPTER 3 Examples

In this chapter ...

This section provides examples of using REST API resources to do a variety of different tasks, including working with objects, organization information, and queries.

For complete reference information on REST API resources, see Reference on page 125.

Getting Information About My Organization

The examples in this section use REST API resources to retrieve organization-level information, such as a list of all objects available in your organization.

IN THIS SECTION:

List Available REST API Versions

Use the Versions resource to list summary information about each REST API version currently available, including the version, label, and a link to each version's root. You don't need authentication to retrieve the list of versions.

Example usage

```
curl https://MyDomainName.my.salesforce.com/services/data/ -H "Authorization: Bearer
token"
```

Example request body

none required

Example JSON response body

```
[
    {
        "label" : "Spring '11",
        "url" : "/services/data/v21.0",
        "version" : "21.0"
    },
    {
        "label" : "Summer '11",
        "url" : "/services/data/v22.0",
        "version" : "22.0"
    },
    {
        "label" : "Winter '12",
        "url" : "/services/data/v23.0",
```

```
        "version" : "23.0"
    }
    ...
]
```


Example XML response body


```
<?xml version="1.0" encoding="UTF-8"?>
<Versions>
    <Version>
        <label>Spring &apos;11</label>
        <url>/services/data/v21.0</url>
        <version>21.0</version>
    </Version>
    <Version>
        <label>Summer &apos;11</label>
        <url>/services/data/v22.0</url>
        <version>22.0</version>
    </Version><Version>
        <label>Winter &apos;12</label>
        <url>/services/data/v23.0</url>
        <version>23.0</version>
    </Version>
    ...
</Versions>
```


SEE ALSO:

Versions

List Org Limits

Use the Limits resource to list your org limits.

- `Max` is the limit for the org.
- `Remaining` is the number of calls or events left for the org.

Example usage

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/limits/ -H
"Authorization: Bearer token" -H "X-PrettyPrint:1"
```

Example request body

none required

Example response body


```
{
    "ActiveScratchOrgs": {
        "Max": 3,
        "Remaining": 3
    },
    "AnalyticsExternalDataSizeMB": {
        "Max": 40960,
        "Remaining": 40960
```

```
},
"ConcurrentAsyncGetReportInstances": {
    "Max": 200,
    "Remaining": 200
},
"ConcurrentEinsteinDataInsightsStoryCreation": {
    "Max": 5,
    "Remaining": 5
},
"ConcurrentEinsteinDiscoveryStoryCreation": {
    "Max": 2,
    "Remaining": 2
},
"ConcurrentSyncReportRuns": {
    "Max": 20,
    "Remaining": 20
},
"DailyAnalyticsDataflowJobExecutions": {
    "Max": 60,
    "Remaining": 60
},
"DailyAnalyticsUploadedFilesSizeMB": {
    "Max": 51200,
    "Remaining": 51200
},
"DailyFunctionsApiCallLimit" : {
  "Max" : 235000,
  "Remaining" : 235000
},
"DailyApiRequests": {
    "Max": 5000,
    "Remaining": 4937
},
"DailyAsyncApexExecutions": {
    "Max": 250000,
    "Remaining": 250000
},
"DailyAsyncApexTests": {
    "Max": 500,
    "Remaining": 500
},
"DailyBulkApiBatches": {
    "Max": 15000,
    "Remaining": 15000
},
"DailyBulkV2QueryFileStorageMB": {
    "Max": 976562,
    "Remaining": 976562
},
"DailyBulkV2QueryJobs": {
    "Max": 10000,
    "Remaining": 10000
},
"DailyDeliveredPlatformEvents" : {
```

```
    "Max" : 10000,
    "Remaining" : 10000
  },
  "DailyDurableGenericStreamingApiEvents": {
      "Max": 10000,
      "Remaining": 10000
  },
  "DailyDurableStreamingApiEvents": {
      "Max": 10000,
      "Remaining": 10000
  },
  "DailyEinsteinDataInsightsStoryCreation": {
      "Max": 1000,
      "Remaining": 1000
  },
  "DailyEinsteinDiscoveryPredictAPICalls": {
      "Max": 50000,
      "Remaining": 50000
  },
  "DailyEinsteinDiscoveryPredictionsByCDC": {
      "Max": 5000000,
      "Remaining": 5000000
  },
  "DailyEinsteinDiscoveryStoryCreation": {
      "Max": 100,
      "Remaining": 100
  },
  "DailyGenericStreamingApiEvents": {
      "Max": 10000,
      "Remaining": 10000
  },
  "DailyScratchOrgs": {
      "Max": 6,
      "Remaining": 6
  },
  "DailyStandardVolumePlatformEvents": {
      "Max": 10000,
      "Remaining": 10000
  },
  "DailyStreamingApiEvents": {
      "Max": 10000,
      "Remaining": 10000
  },
  "DailyWorkflowEmails": {
      "Max": 100000,
      "Remaining": 100000
  },
  "DataStorageMB": {
      "Max": 1024,
      "Remaining": 1024
  },
  "DurableStreamingApiConcurrentClients": {
      "Max": 20,
      "Remaining": 20
```

```
},
"FileStorageMB": {
    "Max": 1024,
    "Remaining": 1024
},
"HourlyAsyncReportRuns": {
    "Max": 1200,
    "Remaining": 1200
},
"HourlyDashboardRefreshes": {
    "Max": 200,
    "Remaining": 200
},
"HourlyDashboardResults": {
    "Max": 5000,
    "Remaining": 5000
},
"HourlyDashboardStatuses": {
    "Max": 999999999,
    "Remaining": 999999999
},
"HourlyLongTermIdMapping": {
    "Max": 100000,
    "Remaining": 100000
},
"HourlyManagedContentPublicRequests": {
    "Max": 50000,
    "Remaining": 50000
},
"HourlyODataCallout": {
    "Max": 20000,
    "Remaining": 20000
},
"HourlyPublishedPlatformEvents": {
    "Max": 50000,
    "Remaining": 50000
},
"HourlyPublishedStandardVolumePlatformEvents": {
    "Max": 1000,
    "Remaining": 1000
},
"HourlyShortTermIdMapping": {
    "Max": 100000,
    "Remaining": 100000
},
"HourlySyncReportRuns": {
    "Max": 500,
    "Remaining": 500
},
"HourlyTimeBasedWorkflow": {
    "Max": 1000,
    "Remaining": 1000
},
"MassEmail": {
```

```
            "Max": 5000,
            "Remaining": 5000
        },
        "MonthlyEinsteinDiscoveryStoryCreation": {
            "Max": 500,
            "Remaining": 500
        },
        "Package2VersionCreates": {
            "Max": 6,
            "Remaining": 6
        },
        "Package2VersionCreatesWithoutValidation": {
            "Max": 500,
            "Remaining": 500
        },
        "PermissionSets": {
            "Max": 1500,
            "Remaining": 1499,
            "CreateCustom": {
                "Max": 1000,
                "Remaining": 999
            }
        },
        "PrivateConnectOutboundCalloutHourlyLimitMB": {
            "Max": 0,
            "Remaining": 0
        },
        "SingleEmail": {
            "Max": 5000,
            "Remaining": 5000
        },
        "StreamingApiConcurrentClients": {
            "Max": 20,
            "Remaining": 20
        }
}
```

List Available REST Resources

Use the Resources by Version resource to list the resources available for the specified API version. This provides the name and URI of each additional resource.

Example

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/ -H "Authorization:
Bearer token"
```

Example request body

none required

Example JSON response body

```
{
   "tooling" : "/services/data/v60.0/tooling",
   "metadata" : "/services/data/v60.0/metadata",
```

```
   "eclair" : "/services/data/v60.0/eclair",
   "folders" : "/services/data/v60.0/folders",
   "prechatForms" : "/services/data/v60.0/prechatForms",
   "contact-tracing" : "/services/data/v60.0/contact-tracing",
   "jsonxform" : "/services/data/v60.0/jsonxform",
   "chatter" : "/services/data/v60.0/chatter",
   "payments" : "/services/data/v60.0/payments",
   "tabs" : "/services/data/v60.0/tabs",
   "appMenu" : "/services/data/v60.0/appMenu",
   "quickActions" : "/services/data/v60.0/quickActions",
   "queryAll" : "/services/data/v60.0/queryAll",
   "commerce" : "/services/data/v60.0/commerce",
   "wave" : "/services/data/v60.0/wave",
   "iot" : "/services/data/v60.0/iot",
   "analytics" : "/services/data/v60.0/analytics",
   "search" : "/services/data/v60.0/search",
   "smartdatadiscovery" : "/services/data/v60.0/smartdatadiscovery",
   "identity" : "https://MyDomainName.my.salesforce.com/id/
      00DRO0000008aXd2AI/005RO000000HfnkYAC",
   "composite" : "/services/data/v60.0/composite",
   "parameterizedSearch" : "/services/data/v60.0/parameterizedSearch",
   "fingerprint" : "/services/data/v60.0/fingerprint",
   "theme" : "/services/data/v60.0/theme",
   "nouns" : "/services/data/v60.0/nouns",
   "domino" : "/services/data/v60.0/domino",
   "event" : "/services/data/v60.0/event",
   "serviceTemplates" : "/services/data/v60.0/serviceTemplates",
   "recent" : "/services/data/v60.0/recent",
   "connect" : "/services/data/v60.0/connect",
   "licensing" : "/services/data/v60.0/licensing",
   "limits" : "/services/data/v60.0/limits",
   "process" : "/services/data/v60.0/process",
   "dedupe" : "/services/data/v60.0/dedupe",
   "async-queries" : "/services/data/v60.0/async-queries",
   "query" : "/services/data/v60.0/query",
   "jobs" : "/services/data/v60.0/jobs",
   "localizedvalue" : "/services/data/v60.0/localizedvalue",
   "mobile" : "/services/data/v60.0/mobile",
   "emailConnect" : "/services/data/v60.0/emailConnect",
   "consent" : "/services/data/v60.0/consent",
   "tokenizer" : "/services/data/v60.0/tokenizer",
   "compactLayouts" : "/services/data/v60.0/compactLayouts",
   "sobjects" : "/services/data/v60.0/sobjects",
   "actions" : "/services/data/v60.0/actions",
   "support" : "/services/data/v60.0/support"
}
```

Example XML response body

```
<?xml version="1.0" encoding="UTF-8"?>
<urls>
   <tooling>/services/data/v60.0/tooling</tooling>
   <metadata>/services/data/v60.0/metadata</metadata>
   <eclair>/services/data/v60.0/eclair</eclair>
   <folders>/services/data/v60.0/folders</folders>
```

```
    <prechatForms>/services/data/v60.0/prechatForms</prechatForms>
    <contact-tracing>/services/data/v60.0/contact-tracing</contact-tracing>
    <jsonxform>/services/data/v60.0/jsonxform</jsonxform>
    <chatter>/services/data/v60.0/chatter</chatter>
    <payments>/services/data/v60.0/payments</payments>
    <tabs>/services/data/v60.0/tabs</tabs>
    <appMenu>/services/data/v60.0/appMenu</appMenu>
    <quickActions>/services/data/v60.0/quickActions</quickActions>
    <queryAll>/services/data/v60.0/queryAll</queryAll>
    <commerce>/services/data/v60.0/commerce</commerce>
    <wave>/services/data/v60.0/wave</wave>
    <iot>/services/data/v60.0/iot</iot>
    <analytics>/services/data/v60.0/analytics</analytics>
    <search>/services/data/v60.0/search</search>
    <smartdatadiscovery>/services/data/v60.0/smartdatadiscovery</smartdatadiscovery>
    <identity>https://MyDomainName.my.salesforce.com/id/
       00DRO0000008aXd2BI/005RO000000HfnkYAB</identity>
    <composite>/services/data/v60.0/composite</composite>
    <parameterizedSearch>/services/data/v60.0/parameterizedSearch</parameterizedSearch>
    <fingerprint>/services/data/v60.0/fingerprint</fingerprint>
    <theme>/services/data/v60.0/theme</theme>
    <nouns>/services/data/v60.0/nouns</nouns>
    <domino>/services/data/v60.0/domino</domino>
    <event>/services/data/v60.0/event</event>
    <serviceTemplates>/services/data/v60.0/serviceTemplates</serviceTemplates>
    <recent>/services/data/v60.0/recent</recent>
    <connect>/services/data/v60.0/connect</connect>
    <licensing>/services/data/v60.0/licensing</licensing>
    <limits>/services/data/v60.0/limits</limits>
    <process>/services/data/v60.0/process</process>
    <dedupe>/services/data/v60.0/dedupe</dedupe>
    <async-queries>/services/data/v60.0/async-queries</async-queries>
    <query>/services/data/v60.0/query</query>
    <jobs>/services/data/v60.0/jobs</jobs>
    <localizedvalue>/services/data/v60.0/localizedvalue</localizedvalue>
    <mobile>/services/data/v60.0/mobile</mobile>
    <emailConnect>/services/data/v60.0/emailConnect</emailConnect>
    <consent>/services/data/v60.0/consent</consent>
    <tokenizer>/services/data/v60.0/tokenizer</tokenizer>
    <compactLayouts>/services/data/v60.0/compactLayouts</compactLayouts>
    <sobjects>/services/data/v60.0/sobjects</sobjects>
    <actions>/services/data/v60.0/actions</actions>
    <support>/services/data/v60.0/support</support>
</urls>
```


Further Information

For information about the identity resource, see Identity URLs.

For the other resources, see Reference .

Get a List of Objects

Use the Describe Global resource to list the objects available in your org and available to the logged-in user. This resource also returns the org encoding, as well as maximum batch size permitted in queries.

Example usage

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/ -H
"Authorization: Bearer token"
```

Example request body

none required

Example response body


```
{
  "encoding" : "UTF-8",
  "maxBatchSize" : 200,
  "sobjects" : [ {
    "activateable" : false,
    "custom" : false,
    "customSetting" : false,
    "createable" : true,
    "deletable" : true,
    "deprecatedAndHidden" : false,
    "feedEnabled" : true,
    "keyPrefix" : "001",
    "label" : "Account",
    "labelPlural" : "Accounts",
    "layoutable" : true,
    "mergeable" : true,
    "mruEnabled" : true,
    "name" : "Account",
    "queryable" : true,
    "replicateable" : true,
    "retrieveable" : true,
    "searchable" : true,
    "triggerable" : true,
    "undeletable" : true,
    "updateable" : true,
    "urls" : {
      "sobject" : "/services/data/v60.0/sobjects/Account",
      "describe" : "/services/data/v60.0/sobjects/Account/describe",
      "rowTemplate" : "/services/data/v60.0/sobjects/Account/{ID}"
    },
  },
  ...
  ]
}
```


Get a List of Objects If Metadata Has Changed

Use the Describe Global resource and the `If-Modified-Since` HTTP header to determine if an object's metadata has changed.

You can include the `If-Modified-Since` header with a date in `EEE, dd MMM yyyy HH:mm:ss z` format when you use the Describe Global resource. If you do, response metadata is returned only if an available object's metadata has changed since the provided date. If no metadata has been modified since the provided date, a `304 Not Modified` status code is returned with no response body.

The following example assumes that no changes have been made to objects after March 23, 2015.

Example Describe Global request

`/services/data/v60.0/sobjects`

Example `If-Modified-Since` header used with request

`If-Modified-Since: Tue, 23 Mar 2015 00:00:00 GMT`

Example response body

No response body returned

Example response status code

```
HTTP/1.1 304 Not Modified
Date: Wed, 25 Jul 2015 00:05:46 GMT
```

If changes to an object were made after March 23, 2015, the response body contains metadata for all available objects. For an example, see Get a List of Objects.

Working with Object Metadata

The examples in this section use REST API resources to retrieve object metadata information. For modifying or creating object metadata information, see the *Metadata API Developer Guide*.

IN THIS SECTION:

Get Metadata for an Object

Use the sObject Basic Information resource to get metadata for an object.

Example for getting Account metadata

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/ -H
"Authorization: Bearer token"
```

Example request body for getting Account metadata

none required

Example response body for getting Account metadata

```
{
  "objectDescribe" :
  {
    "name" : "Account",
    "updateable" : true,
    "label" : "Account",
    "keyPrefix" : "001",

    ...

    "replicateable" : true,
    "retrieveable" : true,
    "undeletable" : true,
    "triggerable" : true
  },
  "recentItems" :
  [
    {
      "attributes" :
      {
        "type" : "Account",
        "url" : "/services/data/v60.0/sobjects/Account/001D000000INjVeIAL"
      },
      "Id" : "001D000000INjVeIAL",
      "Name" : "asdasdasd"
    },

    ...

  ]
}
```

To get a complete description of an object, including field names and their metadata, see Get a List of Objects.

Get Field and Other Metadata for an Object

Use the sObject Describe resource to retrieve all the metadata for an object, including information about each field, URLs, and child relationships.

Example

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/describe/
-H "Authorization: Bearer token"
```

Example request body

none required

Example response body

```
{
  "name" : "Account",
  "fields" :
```

```
  [
    {
      "length" : 18,
      "name" : "Id",
      "type" : "id",
      "defaultValue" : {    "value" : null  },
      "updateable" : false,
      "label" : "Account ID",
      ...
    },

    ...

  ],


  "updateable" : true,
  "label" : "Account",
  "keyPrefix" : "001",
  "custom" : false,

  ...

  "urls" :
  {
    "uiEditTemplate" : "https://MyDomainName.my.salesforce.com/{ID}/e",
    "sobject" : "/services/data/v60.0/sobjects/Account",
    "uiDetailTemplate" : "https://MyDomainName.my.salesforce.com/{ID}",
    ...
  },

  "childRelationships" :
  [
    {
      "field" : "ParentId",
      "deprecatedAndHidden" : false,
      ...
    },

    ....

  ],

  "createable" : true,
  "customSetting" : false,
  ...
}
```

For more information about the items in the request body, see DescribesObjectResult in the *SOAP API Developers Guide*.

Get Object Metadata Changes

Use the sObject Describe resource and the `If-Modified-Since` HTTP header to determine if object metadata has changed.

You can include the `If-Modified-Since` header with a date in `EEE, dd MMM yyyy HH:mm:ss z` format when you use the sObject Describe resource. If you do, response metadata will only be returned if the object metadata has changed since the provided date. If the metadata has not been modified since the provided date, a `304 Not Modified` status code is returned, with no response body.

The following example assumes that no changes, such as new custom fields, have been made to the Merchandise__c object after July 3rd, 2013.

Example sObject Describe request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Merchandise__c/describe
 -H "Authorization: Bearer token" -H "If-Modified-Since: Wed, 3 Jul 2013 19:43:31 GMT"
```

Example response body

No response body returned

Example response status code

```
HTTP/1.1 304 Not Modified
Date: Fri, 12 Jul 2013 05:03:24 GMT
```

If there were changes to Merchandise__c made after July 3rd, 2013, the response body would contain the metadata for Merchandise__c. See Get Field and Other Metadata for an Object for an example.

Working with Records

The examples in this section use REST API resources to create, retrieve, update, and delete records, along with other record-related operations.

IN THIS SECTION:

Create a Record

Use the sObject Basic Information resource to create new records. You supply the required field values in the request data, and send the request using the POST HTTP method. The response body contains the ID of the new record if the call is successful.

Update a Record

You use the sObject Rows resource to update records. Provide the updated record information in your request data and use the PATCH method of the resource with a specific record ID to update that record. Records in a single file must be of the same object type.

Delete a Record

Use the sObject Rows resource to delete records. Specify the record ID and use the DELETE method of the resource to delete a record.

Get Field Values from a Standard Object Record

You use the GET method of the sObject Rows resource to retrieve field values from a record.

Get Field Values from an External Object Record by Using the Salesforce ID

You use the sObject Rows resource to retrieve field values from a record. Specify the fields you want to retrieve in the `fields` parameter and use the GET method of the resource.

Get Field Values from an External Object Record by Using the External ID Standard Field

You use the sObject Rows resource to retrieve field values from a record. Specify the fields you want to retrieve in the `fields` parameter and use the GET method of the resource.

Get a Record Using an External ID

You can use the GET method of the sObject Rows by External ID resource to get records with a specific external ID.

Insert or Update (Upsert) a Record Using an External ID

You can use the sObject Rows by External ID resource to create records or update existing records (upsert) based on the value of a specified external ID field.

Traverse Relationships with Friendly URLs

You can traverse relationship fields in standard and custom objects by constructing friendly URLs using the sObject Relationship resource. This approach allows you to directly access records associated by relationships. Using friendly URLs is an easier alternative to accessing records by obtaining object IDs from relationship fields and then inspecting the associated object ID record.

Get a List of Deleted Records Within a Given Timeframe

Use the sObject Get Deleted resource to get a list of deleted records for the specified object. Specify the date and time range within which the records for the given object were deleted. Deleted records are written to a delete log (that is periodically purged), and will be filtered out of most operations, such as sObject Rows or Query (although QueryAll will include deleted records in results).

Get a List of Updated Records Within a Given Timeframe

Use the sObject Get Updated resource to get a list of updated (modified or added) records for the specified object. Specify the date and time range within which the records for the given object were updated.

Create a Record

Use the sObject Basic Information resource to create new records. You supply the required field values in the request data, and send the request using the POST HTTP method. The response body contains the ID of the new record if the call is successful.

The following example request creates a new Account record, with the field values for the new record provided in `newaccount.json`. Only the name field is specified in this example, but you could also provide values for other Account fields.

Example for creating a new Account

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/ -H
"Authorization: Bearer token" -H "Content-Type: application/json" -d "@newaccount.json"
```

Example request body `newaccount.json` file for creating a new Account

```
{
  "Name" : "Express Logistics and Transport"
}
```

Example response body after successfully creating a new Account

```
{
  "id" : "001D000000IqhSLIAZ",
  "errors" : [ ],
  "success" : true
}
```

Update a Record

You use the sObject Rows resource to update records. Provide the updated record information in your request data and use the PATCH method of the resource with a specific record ID to update that record. Records in a single file must be of the same object type.

In the following example, the Billing City within an Account is updated. The updated record information is provided in `patchaccount.json`.

Example for updating an Account object

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/001D000000INjVe
 -H "Authorization: Bearer token" -H "Content-Type: application/json" -d
@patchaccount.json -X PATCH
```

Example request body `patchaccount.json` file for updating fields in an Account object

```
{
    "BillingCity" : "San Francisco"
}
```

Example response body for updating fields in an Account object

none returned

Error response

See Status Codes and Error Responses on page 17.

The following example uses Java and HttpClient to update a record using REST API. Note that there is no PatchMethod in HttpClient, so PostMethod is overridden to return “PATCH” as its method name. This example assumes the resource URL has been passed in and contains the object name and record ID.

```
public static void patch(String url, String sid) throws IOException {
  PostMethod m = new PostMethod(url) {
    @Override public String getName() { return "PATCH"; }
  };

  m.setRequestHeader("Authorization", "OAuth " + sid);

  Map<String, Object> accUpdate = new HashMap<String, Object>();
  accUpdate.put("Name", "Patch test");
  ObjectMapper mapper = new ObjectMapper();
  m.setRequestEntity(new StringRequestEntity(mapper.writeValueAsString(accUpdate),
"application/json", "UTF-8"));

  HttpClient c = new HttpClient();
  int sc = c.executeMethod(m);
  System.out.println("PATCH call returned a status code of " + sc);
  if (sc > 299) {
    // deserialize the returned error message
    List<ApiError> errors = mapper.readValue(m.getResponseBodyAsStream(), new
TypeReference<List<ApiError>>() {} );
    for (ApiError e : errors)
      System.out.println(e.errorCode + " " + e.message);
  }
}

private static class ApiError {
  public String errorCode;
  public String message;
  public String [] fields;
}
```

If you use an HTTP library that doesn't allow overriding or setting an arbitrary HTTP method name, you can send a POST request and provide an override to the HTTP method via the query string parameter `_HttpMethod`. In the PATCH example, you can replace the PostMethod line with one that doesn't use override:

```
PostMethod m = new PostMethod(url + "?_HttpMethod=PATCH");
```

SEE ALSO:

sObject Rows

Conditional Request Headers

Delete a Record

Use the sObject Rows resource to delete records. Specify the record ID and use the DELETE method of the resource to delete a record.

Example for deleting an Account record

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/001D000000INjVe
 -H "Authorization: Bearer token" -X DELETE
```

Example request body

None needed

Example response body

None returned

Get Field Values from a Standard Object Record

You use the GET method of the sObject Rows resource to retrieve field values from a record.

You can specify the fields you want to retrieve with the optional `fields` parameter. If you specify fields that don't exist or are inaccessible to you by field-level security, a 400 error response is returned.

If you don't use the `fields` parameter, the request retrieves all standard and custom fields from the record. These retrieved fields are the same as the fields returned by an sObject Describe request for the object. Fields that are inaccessible to you by field-level security are not returned in the response body.

In the following example, the Account Number and Billing Postal Code are retrieved from an Account.

Example for retrieving values from fields on an Account object

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/001D000000INjVe
?fields=AccountNumber,BillingPostalCode -H "Authorization: Bearer token"
```

Example request body

None required

Example response body

```
{
    "AccountNumber" : "CD656092",
    "BillingPostalCode" : "27215",
}
```

Get Field Values from an External Object Record by Using the Salesforce ID

You use the sObject Rows resource to retrieve field values from a record. Specify the fields you want to retrieve in the `fields` parameter and use the GET method of the resource.

In the following example, the `Country__c` custom field is retrieved from an external object that's associated with a non-high-data-volume external data source.

Example for retrieving values from fields on the Customer external object

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Customer__x/x01D0000000002RIAQ?fields=Country__c
 -H "Authorization: Bearer token"
```

Example request body

None required

Example response body

```
{
  "attributes" : {
    "type" : "Customer__x",
    "url" : "/services/data/v60.0/sobjects/Customer__x/x01D0000000002RIAQ"
  },
  "Country__c" : "Argentina",
  "Id" : "x01D0000000002RIAQ"
}
```

Get Field Values from an External Object Record by Using the External ID Standard Field

You use the sObject Rows resource to retrieve field values from a record. Specify the fields you want to retrieve in the `fields` parameter and use the GET method of the resource.

In the following example, the `Country__c` custom field is retrieved from an external object. Notice that the *id* (CACTU) isn't a Salesforce ID. Instead, it's the External ID standard field of the external object.

Example for retrieving values from fields on the Customer external object

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Customer__x/CACTU?fields=Country__c
 -H "Authorization: Bearer token"
```

Example request body

None required

Example response body

```
{
  "attributes" : {
    "type" : "Customer__x",
    "url" : "/services/data/v60.0/sobjects/Customer__x/CACTU"
  },
  "Country__c" : "Argentina",
```

```
  "ExternalId" : "CACTU"
}
```

Get a Record Using an External ID

You can use the GET method of the sObject Rows by External ID resource to get records with a specific external ID.

The following example assumes there is a Merchandise__c custom object with a MerchandiseExtID__c external ID field.

Example usage for retrieving a Merchandise__c record using an external ID

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Merchandise__c/MerchandiseExtID__c/123
 -H "Authorization: Bearer token"
```

Example request body

none required

Example response body

```
{
    "attributes" : {
        "type" : "Merchandise__c",
        "url" : "/services/data/v60.0/sobjects/Merchandise__c/a00D0000008oWP8IAM"
     },
    "Id" : "a00D0000008oWP8IAM",
    "OwnerId" : "005D0000001KyEIIA0",
    "IsDeleted" : false,
    "Name" : "Example Merchandise",
    "CreatedDate" : "2012-07-12T17:49:01.000+0000",
    "CreatedById" : "005D0000001KyEIIA0",
    "LastModifiedDate" : "2012-07-12T17:49:01.000+0000",
    "LastModifiedById" : "005D0000001KyEIIA0",
    "SystemModstamp" : "2012-07-12T17:49:01.000+0000",
    "Description__c" : "Merch with external ID",
    "Price__c" : 10.0,
    "Total_Inventory__c" : 100.0,
    "Distributor__c" : null,
    "MerchandiseExtID__c" : 123.0
}
```

Insert or Update (Upsert) a Record Using an External ID

You can use the sObject Rows by External ID resource to create records or update existing records (upsert) based on the value of a specified external ID field.

Important: Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

- If the external ID isn't matched, then a new record is created according to the request body.
- If the external ID is matched one time, then the record is updated according to the request body.
- If the external ID is matched multiple times, then a 300 error is reported, and the record isn't created or updated.

The following sections show you how to work with the external ID resource to retrieve records by external ID and upsert records.

Note: In REST API, upsert uses external ids, not record ids. In Apex, however, upsert can be used with both external ids and record ids. Be aware of the difference if you use both REST API and Apex.

Upserting New Records

This example uses the PATCH method to insert a new record. It assumes that an external ID field, “customExtIdField__c,” has been added to Account. It also assumes that an Account record with a customExtIdField value of 11999 doesn’t already exist.

Example for upserting a record that doesn’t yet exist

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/customExtIdField__c/11999
 -H "Authorization: Bearer token" -H "Content-Type: application/json" -d @newrecord.json
 -X PATCH
```

Example JSON request body `newrecord.json` file

```
{
    "Name" : "California Wheat Corporation",
    "Type" : "New Customer"
}
```

Example JSON response

The successful response is:

```
{
    "id" : "00190000001pPvHAAU",
    "errors" : [ ],
    "success" : true,
    "created": true
}
```

The HTTP status code is 201 (Created).

Note: The `created` parameter is present in the response in API version 46.0 and later. It doesn't appear in earlier versions.

Error responses

Incorrect external ID field:

```
{
    "message" : "The requested resource does not exist",
    "errorCode" : "NOT_FOUND"
}
```

For more information, see Status Codes and Error Responses on page 17.

Inserting New Records Using `Id` as the External ID

This example uses the POST method as a special case to insert a record where the `Id` field is treated as the external ID. Because the value of `Id` is `null`, it’s omitted from the request. This pattern is useful when you’re writing code to upsert multiple records by different external IDs and you don’t want to request a separate resource. POST using `Id` is available in API version 37.0 and later.

Example of inserting a record that doesn't yet exist

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/Id -H
 "Authorization: Bearer token" -H "Content-Type: application/json" -d @newrecord.json
-X POST
```

Example JSON request body `newrecord.json` file

```
{
    "Name" : "California Wheat Corporation",
    "Type" : "New Customer"
}
```

Example JSON response

The successful response is:

```
{
    "id" : "001D000000Kv3g5IAB",
    "success" : true,
    "errors" : [ ],
    "created": true
}
```

The HTTP status code is 201 (Created).

Note: The `created` parameter is present in the response in API version 46.0 and later. It doesn't appear in earlier versions.

Upserting Existing Records

This example uses the PATCH method to update an existing record. It assumes that an external ID field, "customExtIdField__c," has been added to Account and an Account record with a customExtIdField value of 11999 exists. The request uses `updates.json` to specify the updated field values.

Example of upserting an existing record

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/customExtIdField__c/11999
 -H "Authorization: Bearer token" -H "Content-Type: application/json" -d @updates.json
 -X PATCH
```

Example JSON request body `updates.json` file

```
{
    "BillingCity" : "San Francisco"
}
```

Example JSON response

In API version 46.0 and later, the HTTP status code is 200 (OK) and the successful response is:

```
{
    "id" : "001D000000Kv3g5IAB",
    "success" : true,
    "errors" : [ ],
    "created": false
}
```

In API version 45.0 and earlier, the HTTP status code is 204 (No Content) and there isn't a response body.

Error responses

If the external ID value isn't unique, an HTTP status code 300 is returned, plus a list of the records that matched the query. For more information about errors, see Status Codes and Error Responses on page 17.

If the external ID field doesn't exist, an error message and code is returned:

```
{
    "message" : "The requested resource does not exist",
    "errorCode" : "NOT_FOUND"
}
```

Upserting Records and Associating with an External ID

If you have an object that references another object using a relationship, you can use REST API to both insert or update a record and reference another object using an external ID.

The following example creates a record and associates it with a parent record via external ID. It assumes the following:

- A Merchandise__c custom object that has an external ID field named MerchandiseExtID__c.
- A Line_Item__c custom object that has an external ID field named LineItemExtID__c, and a relationship to Merchandise__c.
- A Merchandise__c record exists that has a MerchandiseExtID__c value of 123.
- A Line_Item__c record with a LineItemExtID__c value of 456 does **not** exist. This is the record that gets created and associated to the Merchandise__c record.

Example of upserting a record and referencing a related object

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Line_Item__c/LineItemExtID__c/456
 -H "Authorization: Bearer token" -H "Content-Type: application/json" -d @new.json -X
PATCH
```

Example JSON request body `new.json` file

Notice that the related Merchandise__c record is referenced using the Merchandise__c's external ID field.

```
{
   "Name" : "LineItemCreatedViaExtID",
   "Merchandise__r" :
   {
       "MerchandiseExtID__c" : 123
   }
}
```

Example JSON response

The successful response is:

```
{
    "id" : "a02D0000006YUHrIAO",
    "errors" : [ ],
    "success" : true,
    "created": true
}
```

The HTTP status code is 201 (Created).

Note: The `created` parameter is present in the response in API version 46.0 and later. It doesn't appear in earlier versions.

Error responses

If the external ID value isn't unique, an HTTP status code 300 is returned, plus a list of the records that matched the query. For more information about errors, see Status Codes and Error Responses on page 17.

If the external ID field doesn't exist, an error message and code is returned:

```
{
    "message" : "The requested resource does not exist",
    "errorCode" : "NOT_FOUND"
}
```

You can also use this approach to update existing records. For example, if you created the Line_Item__c shown in the example above, you can try to update the related Merchandise__c using another request.

Example for updating a record

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Line_Item__c/LineItemExtID__c/456
 -H "Authorization: Bearer token" -H "Content-Type: application/json" -d @updates.json
 -X PATCH
```

Example JSON request body `updates.json` file

This assumes another Merchandise__c record exists with a MerchandiseExtID__c value of 333.

```
{
   "Merchandise__r" :
   {
       "MerchandiseExtID__c" : 333
   }
}
```

Example JSON response

In API version 46.0 and later, the HTTP status code is 200 (OK) and the successful response is:

```
{
    "id" : "001D000000Kv3g5IAB",
    "success" : true,
    "errors" : [ ],
    "created": false
}
```

In API version 45.0 and earlier, the HTTP status code is 204 (No Content) and there isn't a response body.

If the relationship type is master-detail and the relationship is set to not allow reparenting, and you try to update the parent external ID, you get an HTTP status code 400 error with an error code of INVALID_FIELD_FOR_INSERT_UPDATE.

SEE ALSO:

sObject Rows by External ID

Traverse Relationships with Friendly URLs

You can traverse relationship fields in standard and custom objects by constructing friendly URLs using the sObject Relationship resource. This approach allows you to directly access records associated by relationships. Using friendly URLs is an easier alternative to accessing records by obtaining object IDs from relationship fields and then inspecting the associated object ID record.

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.

Relationship names follow certain conventions that depend on the direction (parent to child, or child to parent) of the relationship and the name of the related object. The conventions are described in Understanding Relationship Names in the *SOQL and SOSL Reference*.

There are limits to the number of relationship traversals you can make in a single REST API call. These limits are the same as the limits for SOQL, as described in Understanding Relationship Query Limitations in the *SOQL and SOSL Reference*. Keep the following limitations in mind when traversing relationships.

- When specifying child-to-parent relationships, no more than five levels can be traversed. The following traverses two child-to-parent relationships.

  ```
  https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/ChildOfChild__c/record
   id/Child__r/ParentOfChild__r
  ```

- When specifying parent-to-child relationships, no more than one level can be traversed. The following traverses one parent-to-child relationship.

  ```
  https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/ParentOfChild__c/record
   id/Child__r
  ```

Traversing Standard Objects

The standard object Contact contains a relationship field to the Account standard object. The following example retrieves the Account record related to a Contact record.

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Contact/0035e00000PiemmAAB/Account
 -H "Authorization: Bearer token"
```

Example request body for traversing a standard object relationship

none required

Example response body for traversing a standard object simple relationship

```
{
    "attributes": {
        "type": "Account",
        "url": "/services/data/v60.0/sobjects/Account/0015e00000TwULCAA3"
    },
    "Id": "0015e00000TwULCAA3",
    "IsDeleted": false,
    "Name": "relationshipAccountName",
    "PhotoUrl": "/services/images/photo/0015e00000TwULCAA3",
    "OwnerId": "0055e000003E8ooAAC",
    "CreatedDate": "2021-11-06T17:38:40.000+0000",
    "CreatedById": "0055e000003E8ooAAC",
    "LastModifiedDate": "2021-11-06T17:38:40.000+0000",
    "LastModifiedById": "0055e000003E8ooAAC",
    "SystemModstamp": "2021-11-06T17:38:40.000+0000",
    "LastActivityDate": null,
    "LastViewedDate": "2021-11-06T17:40:50.000+0000",
    "LastReferencedDate": "2021-11-06T17:40:50.000+0000"
}
```

Example of traversing a simple relationship

This custom object named Merchandise__c contains a lookup relationship field to a child Distributor__c custom object. The following example retrieves the Distributor__c record related to a Merchandise__c record.

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Merchandise__c/a01D000000INjVe/Distributor__r
 -H "Authorization: Bearer token"
```

Example request body for traversing a simple relationship

none required

Example response body for traversing a simple relationship

```
{
    "attributes" :
    {
        "type" : "Distributor__c",
        "url" : "/services/data/v60.0/sobjects/Distributor__c/a03D0000003DUhcIAG"
    },
    "Id" : "a03D0000003DUhcIAG",
    "OwnerId" : "005D0000001KyEIIA0",
    "IsDeleted" : false,
    "Name" : "Distributor1",
    "CreatedDate" : "2011-12-16T17:43:01.000+0000",
    "CreatedById" : "005D0000001KyEIIA0",
    "LastModifiedDate" : "2011-12-16T17:43:01.000+0000",
    "LastModifiedById" : "005D0000001KyEIIA0",
    "SystemModstamp" : "2011-12-16T17:43:01.000+0000",
    "Location__c" : "San Francisco"
}
```

If no related record is associated with the relationship name, the REST API call fails, because the relationship can't be traversed. Using the previous example, if the Distributor__c field in the Merchandise__c record was set to `null`, the GET call would return a 404 error response.

You can traverse multiple relationships within the same relationship hierarchy in a single REST API call as long as you don't exceed the relationship query limits. If a Line_Item__c custom object is the child in a relationship to a Merchandise__c custom object, and Merchandise__c also has a child Distributor__c custom object, you can access the Distributor__c record starting from the Line_Item__c record using something like the following.

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Line_Item__c/a02D0000006YL7XIAW/Merchandise__r/Distributor__r
 -H "Authorization: Bearer token"
```

Relationship traversal also supports PATCH and DELETE methods for relationships that resolve to a single record. Using the PATCH method, you can update the related record.

Example of using PATCH to update a relationship record

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Merchandise__c/a01D000000INjVe/Distributor__r
 -H "Authorization: Bearer token"  -d @update_info.json -X PATCH
```

Example JSON request body for updating a relationship record contained in `update_info.json`

```
{
    "Location__c" : "New York"
}
```

Example response body for updating relationship record

none returned

Finally, using the DELETE method, you can delete the related record.

Example using DELETE to delete a relationship record

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Merchandise__c/a01D000000INjVe/Distributor__r
 -H "Authorization: Bearer token" -X DELETE
```

Example request body for deleting a relationship record

none required

Example response body for update relationship record

none returned

Traversing Relationships with Multiple Records

You can traverse relationships with multiple records, and get a response that contains the set of records. For relationships that resolve to multiple records, only GET methods are supported.

Example traversing a relationship with multiple records

If we have a custom object named Merchandise__c that contains a master—detail relationship field to a Line_Item__c custom object, the following example retrieves the set of Line_Item__c records related to a Merchandise__c record.

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Merchandise__c/a01D000000INjVe/Line_Items__r
 -H "Authorization: Bearer token"
```

Example request body for traversing a relationship with multiple records

none required

Example response body for traversing a relationship with multiple records

For this example, two Line_Item__c records were retrieved.

```
{
    "done" : true,
    "totalSize" : 2,
    "records" :
    [
        {
            "attributes" :
            {
                "type" : "Line_Item__c",
                "url" : "/services/data/v60.0/sobjects/Line_Item__c/a02D0000006YL7XIAW"

            },
            "Id" : "a02D0000006YL7XIAW",
            "IsDeleted" : false,
            "Name" : "LineItem1",
```

```
            "CreatedDate" : "2011-12-16T17:44:07.000+0000",
            "CreatedById" : "005D0000001KyEIIA0",
            "LastModifiedDate" : "2011-12-16T17:44:07.000+0000",
            "LastModifiedById" : "005D0000001KyEIIA0",
            "SystemModstamp" : "2011-12-16T17:44:07.000+0000",
            "Unit_Price__c" : 9.75,
            "Units_Sold__c" : 10.0,
            "Merchandise__c" : "a00D0000008oLnXIAU",
            "Invoice_Statement__c" : "a01D000000D85hkIAB"
        },
        {
            "attributes" :
            {
                "type" : "Line_Item__c",
               "url" : "/services/data/v60.0/sobjects/Line_Item__c/a02D0000006YL7YIAW"

            },
            "Id" : "a02D0000006YL7YIAW",
            "IsDeleted" : false,
            "Name" : "LineItem2",
            "CreatedDate" : "2011-12-16T18:53:59.000+0000",
            "CreatedById" : "005D0000001KyEIIA0",
            "LastModifiedDate" : "2011-12-16T18:53:59.000+0000",
            "LastModifiedById" : "005D0000001KyEIIA0",
            "SystemModstamp" : "2011-12-16T18:54:00.000+0000",
            "Unit_Price__c" : 8.5,
            "Units_Sold__c" : 8.0,
            "Merchandise__c" : "a00D0000008oLnXIAU",
            "Invoice_Statement__c" : "a01D000000D85hkIAB"
        }
    ]
}
```

The serialized structure for the result data is the same format as result data from executing a SOQL query via REST API. See Query on page 300 for more details on executing SOQL queries via REST API

If no related records are associated with the relationship name, the REST API call returns a 200 response with no record data in the response body. This result is in contrast to the results when traversing an empty relationship to a single record, which returns a 404 error response. This behavior is because the single record case resolves to a REST resource that can be used with PATCH or DELETE methods. In contrast, the multiple record case can only be queried.

If an initial GET request for a relationship with multiple records returns only part of the results, the end of the response contains the field `nextRecordsUrl`. For example, you could get a field like the following at the end of your response.

```
"nextRecordsUrl" : "/services/data/v60.0/query/01gD0000002HU6KIAW-2000"
```

You can request the next batch of records using the provided URL with your instance and session information, and repeat until all records have been retrieved. These requests use `nextRecordsUrl` and don't include any parameters. The final batch of records doesn't have a `nextRecordsUrl` field.

Example usage for retrieving the remaining results

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/query/01gD0000002HU6KIAW-2000
 -H "Authorization: Bearer token"
```

Example request body for retrieving the remaining results

none required

Example response body for retrieving the remaining results

```
{
    "done" : true,
    "totalSize" : 3200,
    "records" : [...]
}
```

Filtering Result Fields

When retrieving records via relationship traversals, you can optionally specify only a subset of the record fields be returned by using the `fields` parameter. Multiple fields are separated by commas. The following example retrieves just the Location__c field from the Distributor__c record associated with a Merchandise__c record:

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Merchandise__c/a01D000000INjVe/Distributor__r?fields=Location__c
 -H "Authorization: Bearer token"
```

The JSON response data would look like the following:

```
{
    "attributes" :
    {
        "type" : "Distributor__c",
        "url" : "/services/data/v60.0/sobjects/Distributor__c/a03D0000003DUhhIAG"
    },
    "Location__c" : "Chicago",
}
```

Similarly, for requests that result in multiple records, you can use a list of fields to specify the fields returned in the record set. For example, assume you have a relationship that was associated with two Line_Item__c records. You want just the Name and Units_Sold__c fields from those records. You could use the following call.

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Merchandise__c/a01D000000INjVe/Line_Items__r?fields=Name,Units_Sold__c
 -H "Authorization: Bearer token"
```

The response data would look like the following.

```
{
    "done" : true,
    "totalSize" : 2,
    "records" :
    [
        {
            "attributes" :
            {
                "type" : "Line_Item__c",
                "url" : "/services/data/v60.0/sobjects/Line_Item__c/a02D0000006YL7XIAW"
            },
            "Name" : "LineItem1",
            "Units_Sold__c" : 10.0
```

```
        },
        {
            "attributes" :
            {
                "type" : "Line_Item__c",
                "url" : "/services/data/v60.0/sobjects/Line_Item__c/a02D0000006YL7YIAW"
            },
            "Name" : "LineItem2",
            "Units_Sold__c" : 8.0
        }
    ]
}
```


If any field listed in the fields parameter set isn't visible to the active user, the REST API call fails. In the previous example, if the Units_Sold_c field was hidden from the active user by field-level security, the call would return a 400 error response.

Get a List of Deleted Records Within a Given Timeframe

Use the sObject Get Deleted resource to get a list of deleted records for the specified object. Specify the date and time range within which the records for the given object were deleted. Deleted records are written to a delete log (that is periodically purged), and will be filtered out of most operations, such as sObject Rows or Query (although QueryAll will include deleted records in results).

Example usage for getting a list of Merchandise__c records that were deleted between May 5th, 2013 and May 10th, 2013

```
curl https://MyDomainName.my/services/data/v60.0/sobjects/Merchandise__c/deleted/
?start=2013-05-05T00%3A00%3A00%2B00%3A00&end=2013-05-10T00%3A00%3A00%2B00%3A00 -H
"Authorization: Bearer token"
```

Example request body

None required

JSON example response body


```
{
    "deletedRecords" :
    [
        {
            "id" : "a00D0000008pQRAIA2",
            "deletedDate" : "2013-05-07T22:07:19.000+0000"
        }
    ],
    "earliestDateAvailable" : "2013-05-03T15:57:00.000+0000",
    "latestDateCovered" : "2013-05-08T21:20:00.000+0000"
}
```


XML example response body


```
<?xml version="1.0" encoding="UTF-8"?>
<Merchandise__c>
    <deletedRecords>
        <deletedDate>2013-05-07T22:07:19.000Z</deletedDate>
        <id>a00D0000008pQRAIA2</id>
    </deletedRecords>
    <earliestDateAvailable>2013-05-03T15:57:00.000Z</earliestDateAvailable>
    <latestDateCovered>2013-05-08T21:20:00.000Z</latestDateCovered>
</Merchandise__c>
```

Get a List of Updated Records Within a Given Timeframe

Use the sObject Get Updated resource to get a list of updated (modified or added) records for the specified object. Specify the date and time range within which the records for the given object were updated.

Example usage for getting a list of Merchandise__c records that were updated between May 6th, 2013 and May 10th, 2013

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Merchandise__c/updated/
?start=2013-05-06T00%3A00%3A00%2B00%3A00&end=2013-05-10T00%3A00%3A00%2B00%3A00 -H
"Authorization: Bearer token"
```

Example request body

None required

JSON example response body


```
{
    "ids" :
    [
        "a00D0000008pQR5IAM",
        "a00D0000008pQRGIA2",
        "a00D0000008pQRFIA2"
    ],
    "latestDateCovered" : "2013-05-08T21:20:00.000+0000"
}
```


XML example response body


```
<?xml version="1.0" encoding="UTF-8"?>
<Merchandise__c>
    <ids>a00D0000008pQR5IAM</ids>
    <ids>a00D0000008pQRGIA2</ids>
    <ids>a00D0000008pQRFIA2</ids>
    <latestDateCovered>2013-05-08T21:20:00.000Z</latestDateCovered>
</Merchandise__c>
```


Delete Lightning Experience Event Series

Use the HTTP DELETE method to remove one or more IsRecurrence2 events in a series. You can remove a single event, all events following and including a specific event, or an entire event series.

Delete a Single Event in a Series

Use the sObject Rows on page 147 resource to delete event records. To delete a single occurrence in a series, specify the event ID and use the DELETE method on page 46 of the resource to delete a record.

Delete Multiple Events or All Events from a Series

To delete all events in a series from a specific occurrence and onwards, specify the event ID and use the DELETE method of the resource to delete a record. When calling this method, IsRecurrence2 must be true and IsRecurrence2Exclusion must be false.

To delete an entire event series, specify the event ID of the first occurrence in the series and use the DELETE method of the resource to delete a record.

Example for deleting multiple events or all events from a series

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Event/00Uxx0000000000/fromThisEventOnwards
 -H "Authorization: Bearer token" -X DELETE
```

Example request body

```
None needed
```

Example response body after successfully deleting events

```
{
success: We're deleting the selected events from the series. Wait for all events to be
 removed.
}
```

Considerations

Deleting from an event onwards doesn't support calls from events that:

- Occurred before the original value of Recurrence2PatternStartDate.
- Are defined as child events.

If the event series originated outside of Salesforce and the event ID of the first occurrence is unavailable, you can't delete all events in a series. Instead, delete events from a specific occurrence onwards.

Working with Searches and Queries

The examples in this section use REST API resources to search and query records using Salesforce Object Search Language (SOSL) and Salesforce Object Query Language (SOQL), and other search APIs. For more information on SOSL and SOQL see the *SOQL and SOSL Reference*.

IN THIS SECTION:

Execute a SOQL Query

Use the Query resource to execute a SOQL query that returns all the results in a single response, or if needed, returns part of the results and a locator used to retrieve the remaining results.

Execute a SOQL Query that Includes Deleted Items

Use the QueryAll resource to execute a SOQL query that includes information about records that have been deleted because of a merge or delete. Use QueryAll rather than Query, because the Query resource will automatically filter out items that have been deleted.

Get Feedback on Query Performance (Beta)

To get feedback on how Salesforce executes your query, report, or list view, use the Query resource along with the `explain` parameter. Salesforce analyzes each query to find the optimal approach to obtain the query results. Depending on the query and query filters, Salesforce uses an index or internal optimization. To return details on how Salesforce optimizes your query, without actually running the query, use the `explain` parameter. Based on the response, decide whether to fine-tune the performance of your query, like adding filters to make the query more selective.

Search for a String

Use the Search resource to execute a SOSL search or use the Parameterized Search resource to execute a simple RESTful search without SOSL.

Get the Default Search Scope and Order

Use the Search Scope and Order resource to retrieve the default global search scope and order for the logged-in user, including any pinned objects in the user's search results page.

Get Search Result Layouts for Objects

Use the Search Result Layouts resource to retrieve the search result layout configuration for each object specified in the query string.

View Relevant Items

Use the Relevant Items resource to get a list of relevant records.

Execute a SOQL Query

Use the Query resource to execute a SOQL query that returns all the results in a single response, or if needed, returns part of the results and a locator used to retrieve the remaining results.

The following query requests the value from `name` fields from all Account records.

Example usage for executing a query

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/query/?q=SELECT+name+from+Account
 -H "Authorization: Bearer token"
```

Example request body for executing a query

none required

Example response body for executing a query

```
{
    "done" : true,
    "totalSize" : 14,
    "records" :
    [
        {
            "attributes" :
            {
                "type" : "Account",
                "url" : "/services/data/v60.0/sobjects/Account/001D000000IRFmaIAH"
            },
            "Name" : "Test 1"
        },
        {
            "attributes" :
            {
                "type" : "Account",
                "url" : "/services/data/v60.0/sobjects/Account/001D000000IomazIAB"
            },
            "Name" : "Test 2"
        },

        ...
```

```
    ]
}
```

Retrieving the Remaining SOQL Query Results

If the initial query returns only part of the results, the end of the response contains a field called `nextRecordsUrl`:

```
"nextRecordsUrl" : "/services/data/v60.0/query/01gD0000002HU6KIAW-2000"
```

In such cases, request the next batch of records and repeat until all records have been retrieved. These requests use `nextRecordsUrl`, and do not include any parameters.

Example usage for retrieving the remaining query results

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/query/01gD0000002HU6KIAW-2000
 -H "Authorization: Bearer token"
```

Example request body for retrieving the remaining query results

none required

Example response body for retrieving the remaining query results

```
{
    "done" : true,
    "totalSize" : 3214,
    "records" : [...]
}
```

Execute a SOQL Query that Includes Deleted Items

Use the QueryAll resource to execute a SOQL query that includes information about records that have been deleted because of a merge or delete. Use QueryAll rather than Query, because the Query resource will automatically filter out items that have been deleted.

The following query requests the value from the `Name` field from all deleted Merchandise__c records, in an organization that has one deleted Merchandise__c record. The same query using Query instead of QueryAll would return no records, because Query automatically filters out any deleted record from the result set.

Example usage for executing a query for deleted Merchandise__c records

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/queryAll/?q=SELECT+Name+from+Merchandise__c+WHERE+isDeleted+=+TRUE
 -H "Authorization: Bearer token"
```

Example request body for executing a query

none required

Example response body for executing a query

```
{
    "done" : true,
    "totalSize" : 1,
    "records" :
    [
        {
            "attributes" :
```

```
            {
                "type" : "Merchandise__c",
              "url" : "/services/data/v60.0/sobjects/Merchandise__c/a00D0000008pQRAIX2"

            },
            "Name" : "Merchandise 1"
        },
    ]
}
```

Retrieving the Remaining SOQL Query Results

If the initial query returns only part of the results, the end of the response will contain a field called `nextRecordsUrl`. For example, you might find this attribute at the end of your query:

```
"nextRecordsUrl" : "/services/data/v60.0/query/01gD0000002HU6KIAW-2000"
```

In such cases, request the next batch of records and repeat until all records have been retrieved. These requests use `nextRecordsUrl`, and do not include any parameters.

Although the `nextRecordsUrl` has `query` in the URL, it still provides remaining results from the initial QueryAll request. The remaining results include deleted records that matched the initial query.

Example usage for retrieving the remaining results

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/query/01gD0000002HU6KIAW-2000
 -H "Authorization: Bearer token"
```

Example request body for retrieving the remaining results

none required

Example response body for retrieving the remaining results

```
{
    "done" : true,
    "totalSize" : 3214,
    "records" : [...]
}
```

Get Feedback on Query Performance (Beta)

To get feedback on how Salesforce executes your query, report, or list view, use the Query resource along with the `explain` parameter. Salesforce analyzes each query to find the optimal approach to obtain the query results. Depending on the query and query filters, Salesforce uses an index or internal optimization. To return details on how Salesforce optimizes your query, without actually running the query, use the `explain` parameter. Based on the response, decide whether to fine-tune the performance of your query, like adding filters to make the query more selective.

Note: This feature is a Beta Service. Customer may opt to try such Beta Service in its sole discretion. Any use of the Beta Service is subject to the applicable Beta Services Terms provided at Agreements and Terms.

The response contains one or more query execution plans, sorted from most optimal to least optimal. The most optimal plan is the plan that's used when the query, report, or list view is executed.

For more details on the fields returned when using `explain`, see the `explain` parameter in Query Options Headers. For more information on making queries more selective, see Working with Very Large SOQL Queries in the *Apex Developer Guide*.

Example: Feedback on Query Performance

The following performance feedback query uses *`Merchandise__c`*:

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/query/?explain=
SELECT+Name+FROM+Merchandise__c+WHERE+CreatedDate+=+TODAY+AND+Price__c+>+10.0
```

The response body for a performance feedback query looks like this:

```
{
  "plans" : [ {
    "cardinality" : 1,
    "fields" : [ "CreatedDate" ],
    "leadingOperationType" : "Index",
    "notes" :  [ {
      "description" : "Not considering filter for optimization because unindexed",
      "fields" : [ "IsDeleted" ],
      "tableEnumOrId" : "Merchandise__c"
    } ],
    "relativeCost" : 0.0,
    "sobjectCardinality" : 3,
    "sobjectType" : "Merchandise__c"
  }, {
    "cardinality" : 1,
    "fields" : [ ],
    "leadingOperationType" : "TableScan",
    "notes" :  [ {
      "description" : "Not considering filter for optimization because unindexed",
      "fields" : [ "IsDeleted" ],
      "tableEnumOrId" : "Merchandise__c"
    } ],
    "relativeCost" : 0.65,
    "sobjectCardinality" : 3,
    "sobjectType" : "Merchandise__c"
  } ]
}
```

This response indicates that Salesforce found two possible execution plans for this query. The first plan uses the CreatedDate index field to improve the performance of this query. The second plan scans all records without using an index. If the query is executed, the first plan is used. Both plans note that there's no secondary optimization for filtering out records marked as deleted because the IsDeleted field isn't indexed.

Search for a String

Use the Search resource to execute a SOSL search or use the Parameterized Search resource to execute a simple RESTful search without SOSL.

Example SOSL Search Using the GET Method

The following example executes a SOSL search for `Acme`. The search string in this example must be URL-encoded.

Example usage

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/search/?q=FIND+%7BAcme%7D
 -H "Authorization: Bearer token"
```

Example request body

None required

Example response body

```
{
  "searchRecords" : [ {
    "attributes" : {
      "type" : "Account",
      "url" : "/services/data/v60.0/sobjects/Account/001D000000IqhSLIAZ"
    },
    "Id" : "001D000000IqhSLIAZ",
  }, {
    "attributes" : {
      "type" : "Account",
      "url" : "/services/data/v60.0/sobjects/Account/001D000000IomazIAB"
    },
    "Id" : "001D000000IomazIAB",
  } ]
}
```

Example Parameterized Search Using the GET Method

The following example executes a parameterized search for `Acme`. The search string in this example must be URL-encoded.

Example usages

Global search for all results containing *Acme*

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/parameterizedSearch/?q=Acme
 -H "Authorization: Bearer token"
```

Account search for results containing *Acme*, returning the id and name fields

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/parameterizedSearch/?q=Acme&sobject=Account&Account.fields=id,name&Account.limit=10
 -H "Authorization: Bearer token"
```

Example request body

None required

Example response body

```
{
  "searchRecords" : [ {
    "attributes" : {
      "type" : "Account",
      "url" : "/services/data/v60.0/sobjects/Account/001D000000IqhSLIAZ"
    },
    "Id" : "001D000000IqhSLIAZ"
  }, {
```

```
    "attributes" : {
      "type" : "Account",
      "url" : "/services/data/v60.0/sobjects/Account/001D000000IomazIAB"
    },
    "Id" : "001D000000IomazIAB"
  } ]
}
```

Example response body with `metadata` parameter

> **Note:** The `metadata` parameter is only returned if the request specified `metadata=LABELS`.

```
{
  "searchRecords" : [ {
    "attributes" : {
      "type" : "Account",
      "url" : "/services/data/v60.0/sobjects/Account/001D000000IqhSLIAZ"
    },
    "Id" : "001D000000IqhSLIAZ",
  }, {
    "attributes" : {
      "type" : "Account",
      "url" : "/services/data/v60.0/sobjects/Account/001D000000IomazIAB"
    },
    "Id" : "001D000000IomazIAB",
  } ],
  "metadata" : {
    "entityetadata" : [ {
      "entityName" : "Account",
      "fieldMetadata" : [ {

        "name" : "Name",
        "label" : "Account Name"
       } ]
    } ]
  }
}
```

Example Parameterized Search Using the POST Method

Execute a parameterized search using the POST method to access all search features available.

Example usage

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/parameterizedSearch
"Authorization: Bearer token" -H "Content-Type: application/json" -d "@search.json"
```

Example request body

None required

Example JSON file

```
{
    "q":"Smith",
```

```
    "fields" : ["id", "firstName", "lastName"],
    "sobjects":[{"fields":["id", "NumberOfEmployees"],
           "name": "Account",
           "limit":20},
          {"name": "Contact"}],
    "in":"ALL",
    "overallLimit":100,
    "defaultLimit":10
}
```

Example response body

```
{
  "searchRecords" : [ {
    "attributes" : {
      "type" : "Contact",
      "url" : "/services/data/v60.0/sobjects/Contact/003xx000004TraiAAC"
    },
    "Id" : "003xx000004TraiAAC",
    "FirstName" : "Smith",
    "LastName" : "Johnson"
  }, {
    "attributes" : {
      "type" : "Account",
      "url" : "/services/data/v60.0/sobjects/Account/001xx000003DHXnAAO"
    },
    "Id" : "001xx000003DHXnAAO",
    "NumberOfEmployees" : 100
  } ]
}
```

SEE ALSO:

Search

Parameterized Search

Get the Default Search Scope and Order

Use the Search Scope and Order resource to retrieve the default global search scope and order for the logged-in user, including any pinned objects in the user's search results page.

In the following example, the default global search scope of the logged-in user consists of the site, idea, case, opportunity, account, and user objects, in the order in which they are returned in the response body.

Example usage

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/search/scopeOrder -H
"Authorization: Bearer token"
```

Example request body

none required

Example response body


```
[
  {
    "type":"Site",
    "url":"/services/data/v60.0/sobjects/Site/describe"
  },
  {
    "type":"Idea",
    "url":"/services/data/v60.0/sobjects/Idea/describe"
  },
  {
    "type":"Case",
    "url":"/services/data/v60.0/sobjects/Case/describe"
  },
  {
    "type":"Opportunity",
    "url":"/services/data/v60.0/sobjects/Opportunity/describe"
  },
  {
    "type":"Account",
    "url":"/services/data/v60.0/sobjects/Account/describe"
  },
  {
    "type":"User",
    "url":"/services/data/v60.0/sobjects/User/describe"
  }
]
```


Get Search Result Layouts for Objects

Use the Search Result Layouts resource to retrieve the search result layout configuration for each object specified in the query string.

Example usage

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/search/layout/?q=Account,Contact,Lead,Asset
 "Authorization: Bearer token"
```

Example request body

None required

Example response body


```
[ { "label" : "Search Results",
    "limitRows" : 25,
    "searchColumns" : [ { "field" : "Account.Name",
          "format" : null,
          "label" : "Account Name",
          "name" : "Name"
        },
        { "field" : "Account.Site",
          "format" : null,
          "label" : "Account Site",
          "name" : "Site"
```

```
      },
      { "field" : "Account.Phone",
        "format" : null,
        "label" : "Phone",
        "name" : "Phone"
      },
      { "field" : "User.Alias",
        "format" : null,
        "label" : "Account Owner Alias",
        "name" : "Owner.Alias"
      }
    ]
},
{ "label" : "Search Results",
  "limitRows" : 25,
  "searchColumns" : [ { "field" : "Contact.Name",
        "format" : null,
        "label" : "Name",
        "name" : "Name"
      },
      { "field" : "Account.Name",
        "format" : null,
        "label" : "Account Name",
        "name" : "Account.Name"
      },
      { "field" : "Account.Site",
        "format" : null,
        "label" : "Account Site",
        "name" : "Account.Site"
      },
      { "field" : "Contact.Phone",
        "format" : null,
        "label" : "Phone",
        "name" : "Phone"
      },
      { "field" : "Contact.Email",
        "format" : null,
        "label" : "Email",
        "name" : "Email"
      },
      { "field" : "User.Alias",
        "format" : null,
        "label" : "Contact Owner Alias",
        "name" : "Owner.Alias"
      }
    ]
},
{ "label" : "Search Results",
  "limitRows" : 25,
  "searchColumns" : [ { "field" : "Lead.Name",
        "format" : null,
        "label" : "Name",
        "name" : "Name"
      },
```

```
        { "field" : "Lead.Title",
          "format" : null,
          "label" : "Title",
          "name" : "Title"
        },
        { "field" : "Lead.Phone",
          "format" : null,
          "label" : "Phone",
          "name" : "Phone"
        },
        { "field" : "Lead.Company",
          "format" : null,
          "label" : "Company",
          "name" : "Company"
        },
        { "field" : "Lead.Email",
          "format" : null,
          "label" : "Email",
          "name" : "Email"
        },
        { "field" : "Lead.Status",
          "format" : null,
          "label" : "Lead Status",
          "name" : "toLabel(Status)"
        },
        { "field" : "Name.Alias",
          "format" : null,
          "label" : "Owner Alias",
          "name" : "Owner.Alias"
        }
      ]
  },
]
```


View Relevant Items

Use the Relevant Items resource to get a list of relevant records.

Example usage for getting a list of the current user's most relevant records

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/relevantItems
 -H "Authorization: Bearer token"
```

Example request body

None required

Example response body


```
[ {
    "apiName" : "Contact",
    "key" : "003",
    "label" : "Contacts",
    "lastUpdatedId" : "135866748",
    "recordIds" : [ "003xx000004TxBA" ]
}, {   "apiName" : "Account",
```

```
    "key" : "001",
    "label" : "Accounts",
    "lastUpdatedId" : "193640553",
    "recordIds" : [ "001xx000003DWsT" ]
}, {
    "apiName" : "User",
    "key" : "005",
    "label" : "Users",
    "lastUpdatedId" : "-199920321",
    "recordIds" : [ "005xx000001Svqw", "005xx000001SvwK", "005xx000001SvwA" ]
}, {  "apiName" : "Case",
    "key" : "069",
    "label" : "Cases",
    "lastUpdatedId" : "1033471693",
    "recordIds" : [ "069xx0000000006", "069xx0000000001", "069xx0000000002" ]
} ]
```


Example usage for filtering the response to certain objects

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/relevantItems?sobjects=Account,User
 -H "Authorization: Bearer token"
```

Example request body

None required

Example response body


```
[ {
    "apiName" : "Account",
    "key" : "001",
    "label" : "Accounts",
    "lastUpdatedId" : "193640553",
    "recordIds" : [ "001xx000003DWsT" ]
}, {
  "apiName" : "User",
    "key" : "005",
    "label" : "Users",
    "lastUpdatedId" : "102959935",
    "recordIds" : [ "005xx000001Svqw", "005xx000001SvwK", "005xx000001SvwA" ]
} ]
```


Example usage for comparing the user’s current list of relevant records to a previous version

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/relevantItems?lastUpdatedId=102959935
 -H "Authorization: Bearer token"
```

Example request body

None required

Example response header

```
lastUpdatedId: 102959935
newResultSetSinceLastQuery: true
```

Example response body

```
[ {
    "apiName" : "User",
    "key" : "003",
    "label" : "Users",
    "lastUpdatedId" : "102959935",
    "recordIds" : [ "003xx000004TxBA" ]
}, {
    "apiName" : "Account",
    "key" : "001",
    "label" : "Accounts",
    "lastUpdatedId" : "193640553",
    "recordIds" : [ "001xx000003DWsT" ]
}, {
    "apiName" : "Case",
    "key" : "005",
    "label" : "Cases",
  "lastUpdatedId" : "1740766611",
    "recordIds" : [ "005xx000001Svqw", "005xx000001SvwA" ]
} ]
```

Example usage for comparing the user's current list of relevant records to a previous version for a particular object

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/relevantItems?mode=MRU&sobjects=Account,Contact&Account.lastUpdatedId=102959935
 -H "Authorization: Bearer token"
```

Example request body

None required

Example response body

```
[ {
    "apiName" : "Account",
    "key" : "001",
    "label" : "Accounts",
    "lastUpdatedId" : "193640553",
    "recordIds" : [ "001xx000003DWsT" ]
} ]
```

SEE ALSO:

sObject Relevant Items

Get an Image from a Rich Text Area Field

Use the sObject Rich Text Image Get resource to retrieve an image from a rich text area field. In this example, we retrieve an image from a custom rich text field of a Lead record called `LeadPhotoRichText__c`. We assume that an image has already been uploaded to this field.

Obtain the Image Reference ID

Before you can retrieve the image with a request, you must first obtain the `refid` from the rich text field. Use the sObject Rows on page 147 resource to retrieve the field information for the Lead record.

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Lead/00Q112222233333
 -H "Authorization: Bearer token"
```

Here’s an example of abridged output from the request.

```
{
    "attributes": {
        "type": "Lead",
        "url": "/services/data/v51.0/sobjects/Lead/00Q112222233333"
    },
    "Id": "00Q112222233333",
    "IsDeleted": false,
    "MasterRecordId": null,
    "LastName": "Smith",
    "FirstName": "John",

    ...

    "LeadPhotoRichText__c": "<img alt=\"johnSmithPhoto\"
src=\"https://MyDomainName.documentforce.com/servlet/rtaImage?eid=a005e000007Dksm&amp;feoid=00N5e00000djU6Y&amp;refid=0EM5e000000B9LQ\"></img>"
}
```

You can see from the LeadPhotoRichText__c field that the `refid` of the image is `0EM5e000000B9LQ`. Use this value in the next step to retrieve the image.

Retrieve the Image

Use the Lead record ID, rich text field name, and image `refid` to retrieve the image. The response returns the image as encoded data with the same file type as it was uploaded with. Save the returned data as an image file with the appropriate file type using the `--output` ***`filename`*** parameter.

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Lead/00Q112222233333/richTextImageFields/LeadPhotoRichText__c/0EMR00000000A8V
 -H "Authorization: Bearer token" --output "LeadPhoto.jpeg"
```

Insert or Update Blob Data

You can use the sObject Basic Information on page 141, sObject Rows on page 147, or sObject Collections on page 393 resources to insert or update binary large objects (blobs) in Salesforce, such as images or PDFs. You can upload files or binary data of any type to any standard object that contains a blob field.

To insert and update blob data, create a multipart request body. The first part of the request body contains non-binary field data, such as the description or name of the new record. The second part contains the binary data of the file you’re uploading. The request body must conform to the MIME multipart content-type standard. For more information, see the W3C Standard for multipart content types.

If you use the sObject Basic Information or sObject Rows resources, the maximum file size for uploads is 2 GB for ContentVersion objects and 500 MB for all other eligible standard objects. If you use the sObject Collections resource, the maximum total size of all files in a single request is 500 MB.

Note: You can insert or update blob data using a non-multipart message, but you are limited to 50 MB of text data or 37.5 MB of base64–encoded data.

These sections provide examples of how to insert or update blob data using a multipart content-type request. The request bodies for these examples use JSON formatting.

Inserting a Document with Blob Data

This example request and request body creates a Document record that contains the binary data of a PDF file. In addition to the binary data of the file itself, the request body also specifies other field data, including the `FolderId`, which is required for the Document object.

If you don't have a Folder record for the new Document record in Salesforce, you must create one using the sObject Basic Information resource before you can follow this example. Make sure the `Type` field of the Folder record is Document.

Tip: After you successfully send the request, you can view the new Document in Salesforce Classic. Salesforce Lightning doesn't display Documents in the user interface.

Example request for creating a Document

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Document/ -H
"Authorization: Bearer token" -H "Content-Type: multipart/form-data;
boundary=\"boundary_string\"" --data-binary @NewDocument.json
```

Example request body for creating a Document

This request body represents the contents of `NewDocument.json`. The binary data for the PDF content has been omitted for brevity and replaced with "Binary data goes here." An actual request contains the full binary content.

```
--boundary_string
Content-Disposition: form-data; name="entity_document";
Content-Type: application/json

{
    "Description" : "Marketing brochure for Q1 2011",
    "Keywords" : "marketing,sales,update",
    "FolderId" : "005D0000001GiU7",
    "Name" : "Marketing Brochure Q1",
    "Type" : "pdf"
}

--boundary_string
Content-Type: application/pdf
Content-Disposition: form-data; name="Body"; filename="2011Q1MktgBrochure.pdf"

Binary data goes here.

--boundary_string--
```

Example response body

On success, the `ID` of the new Document is returned.


```
{
    "id" : "015D0000000N3ZZIA0",
    "errors" : [ ],
    "success" : true
}
```


Example error response


```
{
    "fields" : [ "FolderId" ],
    "message" : "Folder ID: id value of incorrect type: 005D0000001GiU7",
    "errorCode" : "MALFORMED_ID"
}
```


Updating a Document with Blob Data

This example updates an existing Document record. Depending on the contents of the request body, you can update the fields of the record, the binary data associated with it, or both.

If you want to update only the record fields, the request body doesn't require the multipart content type.

Example request for updating binary data in a Document object

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/Document/015D0000000N3ZZIA0
 -H "Authorization: Bearer token" -H "Content-Type: multipart/form-data;
boundary=\"boundary_string\"" --data-binary @UpdateDocument.json -X PATCH
```

Example request body for updating binary data in a Document object

This request body represents the contents of `UpdateDocument.json`. This example updates the binary data of the record, as well as the `Name` and `Keywords` fields. If you want to update only the binary data, you can remove the lines of code with the `Name` and `Keywords` fields.

The binary data for the PDF content has been omitted for brevity and replaced with "Updated document binary goes here." An actual request contains the full binary content.

```
--boundary_string
Content-Disposition: form-data; name="entity_content";
Content-Type: application/json

{
    "Name" : "Marketing Brochure Q1 - Sales",
    "Keywords" : "sales, marketing, first quarter"
}

--boundary_string
Content-Type: application/pdf
Content-Disposition: form-data; name="Body"; filename="2011Q1MktgBrochure.pdf"

Updated document binary data goes here.

--boundary_string--
```

Example response body for updating fields in a Document object

None returned

Error responses

See Status Codes and Error Responses.

Inserting a ContentVersion

This example inserts a new ContentVersion. In addition to the binary data of the file itself, this code also provides values for other fields, such as the `ReasonForChange` and `PathOnClient`. This message contains the `ContentDocumentId` because a ContentVersion is always associated with a ContentDocument.

Tip: The ContentVersion object doesn't support updates. Therefore, you can't update a ContentVersion. You can only insert a new ContentVersion. You can see the results of your changes on the Content tab.

Example usage for inserting a ContentVersion

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/ContentVersion
 -H "Authorization: Bearer token" -H "Content-Type: multipart/form-data;
boundary=\"boundary_string\"" --data-binary @NewContentVersion.json
```

Example request body for inserting a ContentVersion

This request body represents the contents of the file `NewContentVersion.json`. The binary data for the PDF content has been omitted for brevity and replaced with "Binary data goes here." An actual request contains the full binary content.

```
--boundary_string
Content-Disposition: form-data; name="entity_content";
Content-Type: application/json

{
    "ContentDocumentId" : "069D00000000so2",
    "ReasonForChange" : "Marketing materials updated",
    "PathOnClient" : "Q1 Sales Brochure.pdf"
}

--boundary_string
Content-Type: application/octet-stream
Content-Disposition: form-data; name="VersionData"; filename="Q1 Sales Brochure.pdf"

Binary data goes here.

--boundary_string--
```

Example response body for inserting a ContentVersion

```
{
    "id" : "068D00000000pgOIAQ",
    "errors" : [ ],
    "success" : true
}
```

Error responses for inserting a ContentVersion

See Status Codes and Error Responses.

Using sObject Collections to Insert a Collection of Blob Records

This example inserts a collection of new Documents. In addition to the binary data of the files themselves, this code also specifies other field data, such as the `Description` and `Name` for each record in the collection.

Tip: After you successfully send the request, you can view the new Document in Salesforce Classic. Salesforce Lightning doesn't display Documents in the user interface.

Attributes

If you're using sObject Collections with blob data, you must specify certain attribute values in addition to `type` in the request body's attributes map.

Parameter	Description
`binaryPartName`	Required for blob data. A unique identifier for the binary part.
`binaryPartNameAlias`	Required for blob data. The name of the field in which the binary data is inserted or updated.

Example for creating Documents

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/composite/sobjects/ -H
 "Authorization: Bearer token" -H "Content-Type: multipart/form-data;
boundary=\"boundary_string\"" --data-binary @newdocuments.json
```

Example request body for creating Documents

This code is the contents of `newdocuments.json`. The binary data for the PDF content has been omitted for brevity and replaced with "Binary data goes here." An actual request contains the full binary content.

```
--boundary_string
Content-Disposition: form-data; name="collection"
Content-Type: application/json

{
    "allOrNone" : false,
    "records" :
    [
        {
            "attributes" :
            {
                "type" : "Document",
                "binaryPartName": "binaryPart1",
                "binaryPartNameAlias": "Body"
            },
            "Description" : "Marketing Brochure",
            "FolderId" : "005xx000001Svs4AAC",
            "Name" : "Brochure",
            "Type" : "pdf"
        },
        {
            "attributes" :
            {
                "type" : "Document",
                "binaryPartName": "binaryPart2",
```

```
                "binaryPartNameAlias": "Body"
            },
            "Description" : "Pricing Overview",
            "FolderId" : "005xx000001Svs4AAC",
            "Name" : "Pricing",
            "Type" : "pdf"
        }
    ]
}

--boundary_string
Content-Disposition: form-data; name="binaryPart1"; filename="Brochure.pdf"
Content-Type: application/pdf

Binary data goes here.

--boundary_string
Content-Disposition: form-data; name="binaryPart2"; filename="Pricing.pdf"
Content-Type: application/pdf

Binary data goes here.

--boundary_string--
```

Example response body for creating Documents

On success, the IDs of the new Documents are returned.

```
[
    {
        "id": "015xx00000013QjAAI",
        "errors": [],
        "success": true
    },
    {
        "id": "015xx00000013QkAAI",
        "errors": [],
        "success": true
    }
]
```

For more information, see sObject Collections.

Multipart Message Considerations

Following are some considerations for the format of a multipart message when you insert or update blob data.

Boundary String

- Separates the various parts of a multipart request body.
- Must be specified in the `Content-Type` header of multipart request.
- Can be up to 70 characters.
- Can't be a string value that appears anywhere in any part of the request body.

- Includes a two hyphen (--) prefix for the first boundary string.
- Includes a two hyphen (--) suffix for the last boundary string.

Content-Disposition Header

- Required in each part of the request body.
- Must have `form-data` as a value and a `name` attribute.
  - In the non-binary part of the request body, use any value for the `name` attribute.
  - For single documents, in the binary part of the request body, use the `name` attribute to specify the name of the blob data field for the object. For example, the name of the blob data field for the Document object is `Body`.
  - For documents inserted or updated using sObject Collections, use the `name` attribute in each binary part of the request body to specify a unique identifier for that part. These identifiers are referenced by the non-binary part of the request body.
- Must contain a `filename` attribute for the binary part that represents the name of the local file.

Content-Type Header

- Required in each part of the multipart request body.
- Supports the `application/json` and `application/xml` content types for the non-binary part of the multipart request body.
- Supports any content type for the binary part of the multipart request body.

New Line

A new line must be added between the header and the data for each part of the multipart request body. As you can see in the examples, a new line separates the `Content-Type` and `Content-Disposition` headers from the JSON or binary data.

SEE ALSO:

sObject Basic Information

sObject Rows

sObject Collections

Get Blob Data

Get Blob Data

Use the sObject Blob Get resource to get blob data for a given record. To get blob data, a record with blob data must exist in Salesforce.

Only certain standard objects have blob fields, such as Attachment, ContentNote, ContentVersion, Document, Folder, and Note.

Note: The sObject Blob Get resource isn't compatible with Composite API requests, because it returns binary data instead of data in JSON or XML formats. Instead, make individual sObject Blob Get requests to retrieve blob data.

The following example gets the blob data for the Document record that was created in Insert or Update Blob Data on page 73.

Example for retrieving blob data from a Document record

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Document/015D0000000N3ZZIA0/body
 -H "Authorization: Bearer token"
```

Example request body

none required

Example response body

Document body content is returned in binary form. The response content type isn't JSON or XML since the returned data is binary. You can save the returned binary data to a file to store and access it.

SEE ALSO:

Insert or Update Blob Data

Working with Recently Viewed Information

The examples in this section use REST API Query and Recently Viewed resources to programmatically retrieve and update recently viewed record information.

IN THIS SECTION:

View Recently Viewed Records

Use the Recently Viewed Items resource to get a list of recently viewed records.

Mark Records as Recently Viewed

To mark a record as recently viewed using REST API, use the Query resource with a `FOR VIEW` or `FOR REFERENCE` clause. Use SOQL to mark records as recently viewed to ensure that information such as the date and time the record was viewed is correctly set.

View Recently Viewed Records

Use the Recently Viewed Items resource to get a list of recently viewed records.

Example usage for getting the last two most recently viewed records

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/recent/?limit=2 -H
"Authorization: Bearer token"
```

Example request body

none required

Example response body


```
{
    "attributes" :
    {
        "type" : "Account",
        "url" : "/services/data/v60.0/sobjects/Account/a06U000000CelH0IAJ"
    },
    "Id" : "a06U000000CelH0IAJ",
    "Name" : "Acme"
},
{
    "attributes" :
    {
        "type" : "Opportunity",
        "url" : "/services/data/v60.0/sobjects/Opportunity/a06U000000CelGvIAJ"
    },
    "Id" : "a06U000000CelGvIAJ",
```

```
    "Name" : "Acme - 600 Widgets"
}
```

Mark Records as Recently Viewed

To mark a record as recently viewed using REST API, use the Query resource with a `FOR VIEW` or `FOR REFERENCE` clause. Use SOQL to mark records as recently viewed to ensure that information such as the date and time the record was viewed is correctly set.

Use `FOR VIEW` to notify Salesforce when a record is viewed from a custom interface, such as a mobile application or from a custom page. Use `FOR REFERENCE` when a record is referenced from a custom interface. A record is referenced every time a related record is viewed. For more information, see "FOR VIEW" and "FOR REFERENCE" in the *SOQL and SOSL Reference*.

Example usage for executing a query that marks one Account record as recently viewed

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/query/?q=SELECT+Name+FROM+Account+LIMIT+1+FOR+VIEW
 -H "Authorization: Bearer token"
```

Example request body for executing a query

none required

Example response body for executing a query

```
{
    "done" : true,
    "totalSize" : 1,
    "records" :
    [
        {
            "attributes" :
            {
                "type" : "Account",
                "url" : "/services/data/v60.0/sobjects/Account/001D000000IRFmaIAH"
            },
            "Name" : "Acme"
        },

    ]
}
```

SEE ALSO:

Query

Managing User Passwords

The examples in this section use REST API resources to manage user passwords, such as setting or resetting passwords.

IN THIS SECTION:

Manage User Passwords

Use the sObject User Password resource to set, reset, or get information about a user password. Use the HTTP GET method to get password expiration status, the HTTP POST method to set the password, and the HTTP DELETE method to reset the password.

Manage User Passwords

Use the sObject User Password resource to set, reset, or get information about a user password. Use the HTTP GET method to get password expiration status, the HTTP POST method to set the password, and the HTTP DELETE method to reset the password.

The associated session must have permission to access the given user password information. If the session does not have proper permissions, an HTTP error 403 response is returned from these methods.

These methods are available for both users and self-service users. For managing self-service user passwords, use `SelfServiceUser` instead of `User` in the REST API URL.

Here is an example of retrieving the current password expiration status for a user:

Example usage for getting current password expiration status

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/User/005D0000001KyEIIA0/password
 -H "Authorization: Bearer token"
```

Example request body for getting current password expiration status

None required

JSON example response body for getting current password expiration status


```
{
    "isExpired" : false
}
```


XML example response body for getting current password expiration status


```
<Password>
    <isExpired>false</isExpired>
</Password>
```


Example error response if session has insufficient privileges


```
{
    "message" : "You do not have permission to view this record.",
    "errorCode" : "INSUFFICIENT_ACCESS"
}
```


Here is an example of changing the password for a given user:

Example usage for changing a user password

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/User/005D0000001KyEIIA0/password
 -H "Authorization: Bearer token" —H "Content-Type: application/json" —d @newpwd.json
—X POST
```

Contents for file newpwd.json

```
{
    "NewPassword" : "myNewPassword1234"
}
```

Example response for changing a user password

No response body on successful password change, HTTP status code 204 returned.

Example error response if new password does not meet organization password requirements

```
{
    "message" : "Your password must have a mix of letters and numbers.",
    "errorCode" : "INVALID_NEW_PASSWORD"
}
```

And finally, here is an example of resetting a user password:

Example usage for resetting a user password

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/User/005D0000001KyEIIA0/password
 -H "Authorization: Bearer token" —X DELETE
```

Example request body for resetting a user password

None required

JSON example response body for resetting a user password

```
{
    "NewPassword" : "2sv0xHAuM"
}
```

XML example response body for resetting a user password

```
<Result>
    <NewPassword>2sv0xHAuM</NewPassword>
</Result>
```

SEE ALSO:

sObject User Password

Working with Approval Processes and Process Rules

The examples in this section use REST API resources to work with approval processes and process rules.

IN THIS SECTION:

Get a List of All Approval Processes

Use the Process Approvals resource to get information about approvals.

Submit a Record for Approval

Use the Process Approvals resource to submit a record or a collection of records for approval. Each call takes an array of requests.

Approve a Record

Use the Process Approvals resource to approve a record or a collection of records. Each call takes an array of requests. The current user must be an assigned approver.

Reject a Record

Use the Process Approvals resource to reject a record or a collection of records. Each call takes an array of requests. The current user must be an assigned approver.

Bulk Approvals

Use the Process Approvals resource to do bulk approvals. You can specify a collection of different Process Approvals requests to have them all executed in bulk.

Get a List of Process Rules

Use the Process Rules resource to get information about process rules.

Get a Particular Process Rule

Use the Process Rules resource and specify the *sObjectName* and *workflowRuleId* of the rule you want to get the metadata for.

Trigger Process Rules

Use the Process Rules resource to trigger process rules. All rules associated with the specified ID will be evaluated, regardless of the evaluation criteria. All IDs must be for records on the same object.

Get a List of All Approval Processes

Use the Process Approvals resource to get information about approvals.

Example usage

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/process/approvals/ -H
"Authorization: Bearer token"
```

Example request body

none required

Example JSON response body

```
{
  "approvals" : {
   "Account" : [ {
     "description" : null,
     "id" : "04aD00000008Py9",
     "name" : "Account Approval Process",
     "object" : "Account",
     "sortOrder" : 1
   } ]
  }
}
```

Submit a Record for Approval

Use the Process Approvals resource to submit a record or a collection of records for approval. Each call takes an array of requests.

Example usage

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/process/approvals/ -H
"Authorization: Bearer token" -H "Content-Type: application/json" -d @submit.json"
```

Example request body `submit.json` file

In the following example, the record "001D000000I8mIm" is submitted for approval process "PTO_Request_Process" by skipping its entry criteria on behalf of submitter "005D00000015rZy."

```
{
"requests" : [{
"actionType": "Submit",
"contextId": "001D000000I8mIm",
"nextApproverIds": ["005D00000015rY9"],
"comments":"this is a test",
"contextActorId": "005D00000015rZy",
"processDefinitionNameOrId" : "PTO_Request_Process",
"skipEntryCriteria": "true"}]
}
```

Example JSON response body

```
[ {
  "actorIds" : [ "005D00000015rY9IAI" ],
   "entityId" : "001D000000I8mImIAJ",
   "errors" : null,
   "instanceId" : "04gD0000000Cvm5IAC",
   "instanceStatus" : "Pending",
   "newWorkitemIds" : [ "04iD0000000Cw6SIAS" ],
   "success" : true } ]
```

Approve a Record

Use the Process Approvals resource to approve a record or a collection of records. Each call takes an array of requests. The current user must be an assigned approver.

Example usage

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/process/approvals/ -H
"Authorization: Bearer token" -H "Content-Type: application/json" -d @approve.json"
```

Example request body `approve.json` file

```
{
  "requests" : [{
    "actionType" : "Approve",
    "contextId" : "04iD0000000Cw6SIAS",
    "nextApproverIds" : ["005D00000015rY9"],
    "comments" : "this record is approved"}]
}
```

Example JSON response body

```
[ {
  "actorIds" : null,
  "entityId" : "001D000000I8mImIAJ",
  "errors" : null,
  "instanceId" : "04gD0000000CvmAIAS",
```

```
  "instanceStatus" : "Approved",
  "newWorkitemIds" : [ ],
  "success" : true
} ]
```

Reject a Record

Use the Process Approvals resource to reject a record or a collection of records. Each call takes an array of requests. The current user must be an assigned approver.

Example usage

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/process/approvals/ -H
"Authorization: Bearer token" -H "Content-Type: application/json" -d @reject.json"
```

Example request body `reject.json` file

```
{
  "requests" : [{
    "actionType" : "Reject",
    "contextId" : "04iD0000000Cw6cIAC",
    "comments" : "This record is rejected."}]
}
```

Example JSON response body

```
[ {
  "actorIds" : null,
  "entityId" : "001D000000I8mImIAJ",
  "errors" : null,
  "instanceId" : "04gD0000000CvmFIAS",
  "instanceStatus" : "Rejected",
  "newWorkitemIds" : [ ],
  "success" : true
} ]
```

Bulk Approvals

Use the Process Approvals resource to do bulk approvals. You can specify a collection of different Process Approvals requests to have them all executed in bulk.

Example usage

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/process/approvals/ -H
"Authorization: Bearer token" -H "Content-Type: application/json" -d @bulk.json"
```

Example request body `bulk.json` file

```
{
  "requests" :
  [{
    "actionType" : "Approve",
    "contextId" : "04iD0000000Cw6r",
    "comments" : "approving an account"
```

```
    },{
    "actionType" : "Submit",
    "contextId" : "001D000000JRWBd",
    "nextApproverIds" : ["005D00000015rY9"],
    "comments" : "submitting an account"
    },{
    "actionType" : "Submit",
    "contextId" : "003D000000QBZ08",
    "comments" : "submitting a contact"
    }]
}
```

Example JSON response body


```
[ {
  "actorIds" : null,
  "entityId" : "001D000000I8mImIAJ",
  "errors" : null,
  "instanceId" : "04gD0000000CvmZIAS",
  "instanceStatus" : "Approved",
  "newWorkitemIds" : [ ],
  "success" : true
  }, {
  "actorIds" : null,
  "entityId" : "003D000000QBZ08IAH",
  "errors" : null,
  "instanceId" : "04gD0000000CvmeIAC",
  "instanceStatus" : "Approved",
  "newWorkitemIds" : [ ],
  "success" : true
  }, {
  "actorIds" : [ "005D00000015rY9IAI" ],
  "entityId" : "001D000000JRWBdIAP",
  "errors" : null,
  "instanceId" : "04gD0000000CvmfIAC",
  "instanceStatus" : "Pending",
  "newWorkitemIds" : [ "04iD0000000Cw6wIAC" ],
  "success" : true
} ]
```


Get a List of Process Rules

Use the Process Rules resource to get information about process rules.

Example usage

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/process/rules/ -H
"Authorization: Bearer token"
```

Example request body

none required

Example JSON response body


```
{
  "rules" : {
    "Account" : [ {
      "actions" : [ {
        "id" : "01VD0000000D2w7",
        "name" : "ApprovalProcessTask",
        "type" : "Task"
      } ],
      "description" : null,
      "id" : "01QD0000000APli",
      "name" : "My Rule",
      "namespacePrefix" : null,
      "object" : "Account"
    } ]
  }
}
```


Get a Particular Process Rule

Use the Process Rules resource and specify the *sObjectName* and *workflowRuleId* of the rule you want to get the metadata for.

Example usage

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/process/rules/Account/01QD0000000APli
 -H "Authorization: Bearer token"
```

Example request body

none required

Example JSON response body


```
{
  "actions" : [ {
    "id" : "01VD0000000D2w7",
    "name" : "ApprovalProcessTask",
    "type" : "Task"
    } ],
    "description" : null,
    "id" : "01QD0000000APli",
    "name" : "My Rule",
    "namespacePrefix" : null,
    "object" : "Account"
}
```


Trigger Process Rules

Use the Process Rules resource to trigger process rules. All rules associated with the specified ID will be evaluated, regardless of the evaluation criteria. All IDs must be for records on the same object.

Example usage

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/process/rules/ -H
"Authorization: Bearer token" -H "Content-Type: application/json" -d @rules.json
```

Example request body `rules.json` file

```
{
  "contextIds" : [
    "001D000000JRWBd",
    "001D000000I8mIm",
    "001D000000I8aaf"]
}
```

Example JSON response body

```
{
  "errors" : null,
  "success" : true
}
```

Using Event Monitoring

These examples use REST API event monitoring data that contains information useful for assessing org usage trends and user behavior. Event monitoring is accessed through the Lightning Platform SOAP API and REST API by way of the EventLogFile object. Therefore, you can integrate log data with your own back-end storage and data marts to correlate data from multiple orgs and across disparate systems.

Note: For the supported event types that you can use with event monitoring, see Object Reference for Salesforce and Lightning Platform: EventLogFile Object.

- In the unlikely case where no log files are generated for 24 hours, contact Salesforce Customer Support.
- Log data is read only. You can't insert, update, or delete log data.
- To determine which files were generated for your org, use the `EventType` field.
- An event generates log data in real time. However, daily log files are generated during nonpeak hours the day after an event takes place. Therefore, daily log file data is unavailable for at least 1 day after an event. For hourly log files, depending on event delivery and final processing time, expect an event to take place 3–6 hours from the time of the event to be available in the log file. However, it can take longer.
- Log files are generated only when at least one event of a type, represented by the `EventType` field, occurs for the day or hour. If no events took place, the file isn't generated.
- Log files are available for 30 days after their `CreatedDate`in orgs with Event Monitoring licenses, after which they're automatically deleted. In all Developer Edition orgs, log files are available for 1 day.
- All event monitoring logs are exposed to the API through the EventLogFile object. However, there's no access through the user interface, except through the Event Monitoring Analytics app.
- Log files don't count towards your org's data or file storage allocations.
- Event Monitoring log files aren't a system of record for user activity. They're a source of truth, but they aren't durable. During Salesforce site switches, instance refreshes, or unplanned system outages, data loss can occur. For example, if Salesforce moves your production org instance, your event log files can have a gap in data. Salesforce makes commercially reasonable efforts to preserve event log file data integrity and avoid data loss. When Salesforce performs a site switch or instance refresh, it uses an automated process to replicate event logs.

- Hourly event log files are provided for you to review events in your orgs on an accelerated basis. However, it's possible that you don't get all event log data in hourly event log files, especially during site switches, instance refreshes, or unplanned system outages. For complete data, use the daily log files.
- If event transmission failures take too long to recover from, log files are retransmitted to ensure that they're delivered at least one time. As a result, latent log files can sometimes contain duplicate event data. When your application consumes latent log files, make sure that your application handles duplicate event delivery.
- When a daily incremental log file is delivered, a new file replaces the original file with the full set of available logs for that date. To ensure that you're looking at the most recent log file, check the `CreatedDate` field.
- We recommend that you always query the EventLogFile object for new log files to ensure that you also include latent ones. To identify newly created log files, use the EventLogFile `CreatedDate` field. Hourly and daily incremental logs are delivered differently. For details, see Differences Between Hourly and 24-Hour Event Logs.

All queries and examples in this section require the View Event Log Files and API Enabled user permissions. Users with the View All Data permission can also view event monitoring data.

IN THIS SECTION:

Describe Event Monitoring Using REST

Use the sObject Describe resource to retrieve all metadata for an object, including information about fields, URLs, and child relationships.

Example

You can use REST API to describe event log files. Use a GET request like this:

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/EventLogFile/describe
 -H "Authorization: Bearer token"
```

Example raw response

```
{
   "actionOverrides" : [ ],
   "activateable" : false,
   "childRelationships" : [ ],
   "compactLayoutable" : false,
   "createable" : false,
   "custom" : false,
   "customSetting" : false,
   "deletable" : false,
   "deprecatedAndHidden" : false,
   "feedEnabled" : false,
   "fields" : [ {
     "autoNumber" : false,
     "byteLength" : 18,
     "calculated" : false,
     "calculatedFormula" : null,
     "cascadeDelete" : false,
     "caseSensitive" : false,
     "controllerName" : null,
     "createable" : false,
     ...
}
```

Query Event Monitoring Data with REST

Use the Query resource to retrieve field values from a record. Specify the fields you want to retrieve in the fields parameter and use the `GET` method of the resource.

You can use REST API to query event monitoring data. To retrieve event monitoring records based on `LogDate` and `EventType`, use a GET request like this:

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/query?q=SELECT+Id+,+EventType+,+LogFile+
,+LogDate+,+LogFileLength+FROM+EventLogFile+WHERE+
LogDate+>+Yesterday+AND+EventType+=+'API' -H "Authorization: Bearer token"
```

Example raw response

```
{
   "totalSize" : 4,
   "done" : true,
   "records" : [ {
     "attributes" : {
       "type" : "EventLogFile",
       "url" : "/services/data/v60.0/sobjects/EventLogFile/0ATD000000001bROAQ"      }
     "Id" : "0ATD000000001bROAQ",
     "EventType" : "API",
    "LogFile" : "/services/data/v60.0/sobjects/EventLogFile/0ATD000000001bROAQ/LogFile",

     "LogDate" : "2014-03-14T00:00:00.000+0000",
     "LogFileLength" : 2692.0
    }, {
```

```
      "attributes" : {
        "type" : "EventLogFile",
        "url" : "/services/data/v60.0/sobjects/EventLogFile/0ATD000000001SdOAI"     },
        "Id" : "0ATD000000001SdOAI",
        "EventType" : "API",
        "LogFile" :
"/services/data/v60.0/sobjects/EventLogFile/0ATD000000001SdOAI/LogFile",
        "LogDate" : "2014-03-13T00:00:00.000+0000",
        "LogFileLength" : 1345.0
    }, {
        "attributes" : {
         "type" : "EventLogFile",
         "url" : "/services/data/v60.0/sobjects/EventLogFile/0ATD000000003p1OAA"     },
         "Id" : "0ATD000000003p1OAA",
         "EventType" : "API",
         "LogFile" :
"/services/data/v60.0/sobjects/EventLogFile/0ATD000000003p1OAA/LogFile",
         "LogDate" : "2014-06-21T00:00:00.000+0000",
         "LogFileLength" : 605.0    },
 {      "attributes" : {
        "type" : "EventLogFile",
        "url" : "/services/data/v60.0/sobjects/EventLogFile/0ATD0000000055eOAA"     },
        "Id" : "0ATD0000000055eOAA",
        "EventType" : "API",
        "LogFile" :
"/services/data/v60.0/sobjects/EventLogFile/0ATD0000000055eOAA/LogFile",
        "LogDate" : "2014-07-03T00:00:00.000+0000",
        "LogFileLength" : 605.0
      } ]
}
```


Get Event Monitoring Content from a Record

Use the sObject Blob Retrieve resource to retrieve BLOB data for a given record.

Example

You can use REST API to retrieve BLOB data for event monitoring. Use a GET request similar to this:

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/EventLogFile/0ATD000000000pyOAA/LogFile
 -H "Authorization: Bearer token"
```

Example response body

Event monitoring content is returned in binary form. Note that the response content type won't be JSON or XML because the returned data is binary.

```
HTTP/1.1 200 OK
Date: Tue, 06 Aug 2013 16:46:10 GMT
Sforce-Limit-Info: api-usage=135/5000
Content-Type: application/octetstream
Transfer-Encoding: chunked
"EVENT_TYPE", "ORGANIZATION_ID", "TIMESTAMP","USER_ID", "CLIENT_IP",
"URI", "REFERRER_URI", "RUN_TIME"
"URI", "00DD0000000K5xD", "20130728185606.020", "005D0000001REDy",
```

```
"10.0.62.141", "/secur/contentDoor", "https-//login-salesforce-com/",
"11"
"URI", "00DD0000000K5xD", "20130728185556.930", "005D0000001REI0",
"10.0.62.141", "/secur/logout.jsp", "https-//MyDomainName-my-salesforce-com/000/o",
"54"
"URI", "00DD0000000K5xD", "20130728185536.725", "005D0000001REI0",
"10.0.62.141", "/00OD0000001ckx3",
"https-//MyDomainName-my-salesforce-com/00OD0000001ckx3", "93"
```

Download Large Event Log Files Using cURL with REST

You might have some event log files that are larger than your tool can handle. A command line tool such as cURL is one method to download files larger than 100 MB using the sObject Blob Get object

Example: Use the "X-PrettyPrint" header and the "-o" flag to output large files to .csv formats

This command downloads a file onto your machine into your downloads folder.

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/EventLogFile/0AT30000000000uGAA/LogFile
-H "Authorization: Bearer token" -H "X-PrettyPrint:1" -o ~/downloads/outputLogFile.csv
```

We recommend using compression when downloading large event log files. See Compression Headers.

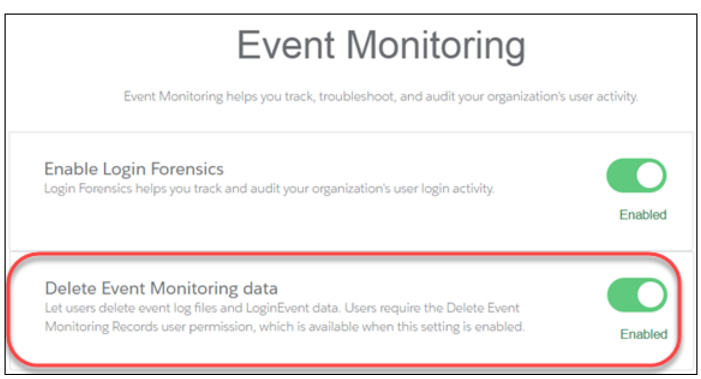
Delete Event Monitoring Data

You can delete event log files that contain a user's log data. Deleting log files helps you comply with data protection and privacy regulations and controls the information that others can access. You can't delete individual rows from event logs. Instead, you must delete the entire log file that contains the user activity.

To delete an event log file, enable deletion in Setup, create a permission set that includes the Delete Event Monitoring Records user permission, and assign this permission set to your users. (Alternatively, you can assign the user permission to a custom profile.) Then those users can query and delete the EventLogFile record by using Query and Delete resources in REST or `delete()` in SOAP.

Note: You can't delete individual rows from event logs. Because event logs are stored in blob format in the database, you must delete the entire log file that contains the user activity.

1. In Setup, in the Quick Find box, enter *Event*, and then select **Event Monitoring Settings**.
2. Enable deletion of event monitoring data. This action is recorded in Setup Audit Trail.



The Delete Event Monitoring Records user permission is now available to assign to a permission set. (Alternatively, you can assign the user permission to a custom profile.)

3. In Setup, in the Quick Find box, enter `Permission`, and then select **Permission Sets**.
4. Create a permission set that includes the Delete Event Monitoring Records user permission, and save the permission set.
5. In Setup, in the Quick Find box, enter `users`, and then select **Users**.
6. Select the user you want to grant permission to delete event monitoring data.
7. In the Permission Set Assignment section for this user, assign the permission set, and click **Save**. This action is recorded in Setup Audit Trail.
   Users assigned this permission set (or any custom profile that includes the Delete Event Monitoring Records user permission) can now delete event monitoring data. The next steps show you how to use the API to delete the data.
8. To locate the logs containing the user activity that you want to delete, query the EventLogFile object. For details, see Query Event Monitoring Data with REST on page 91.
9. Note the IDs of the returned logs.
10. Use the sObject Rows resource to delete records. Specify the record ID, and use the DELETE method. For more information, see Delete a Record on page 46.

Query or View Hourly Event Log Files

To review events in your org on an accelerated basis, get event log files in hourly increments for recent activity. Hourly event log files can give you quicker visibility into security anomalies and custom code performance issues.

EDITIONS

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

USER PERMISSIONS

To access the API and query log files:
- API Enabled AND View Event Log Files

To view event log files:
- View All Data

Examples

Suppose you're a security analyst monitoring for anomalous user behavior. By pulling more frequent updates into your security system, you can be alerted that a suspicious event has taken place within hours, rather than one or two days later.

In another example, let's say you're a developer. You've identified a series of Apex failures in your org, and you want to proactively refactor your Apex code to improve performance. You review hourly log files to pinpoint the issues and fix your code in hours, before your end users start complaining about poor performance.

Considerations

- Hourly event log file integration with the Event Monitoring Analytics app is unavailable.
- Depending on event delivery and final processing time, an event is expected to take three to six hours from the time of the event to be available in the log file. However, it can take longer.
- When delays in processing occur and event logs for a particular hour arrive later, a new log file is created for the event/date/hour and lists only the new events. Use the creation date and an incremental sequence number to identify a new log file. Always use the most recently processed event log file for a particular date. However, if event log files have already been pulled into a third-party application, they could require deduplication in that application.

  If both hourly and daily logs are enabled, daily logs always have a sequence number of 0 because there is only one file per daily interval. CreatedDate indicates when the file was generated. If CreatedDate is after the last event log file download, there are new events to be processed.

  For best practices, use CreatedDate in the WHERE clause to select logs created after the last downloaded event log file. For example, if the last downloaded file was at 12PM 2/1/2018, to find the next log file, use +CreatedDate+>+"2018-02-01T12:00:00Z" in the WHERE clause.

- Your event log files may show a gap in data during site switches, instance refreshes, or unplanned system outages. However, during site switches and instance refreshes, Salesforce makes a commercially reasonable effort to avoid such data loss by using an automated process to replicate event logs.
- In the unlikely case in which no log files are generated for 24 hours, contact Salesforce Support.

IN THIS SECTION:

Query Hourly Event Log Files

You query hourly event log files in the same way you query 24-hour log files.

Suppose you're an administrator. Your Chief Security Officer asks you to identify who modified specific accounts and opportunities in the past two hours. You query the hourly URI event log files using the EventLogFile object to review the page requests and request status. Because EventLogFile also returns 24-hour log files, use this SOQL syntax to filter out the 24-hour log files.

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/query?q=SELECT+Id+,
+EventType+,+Interval+,+LogDate+,+LogFile+FROM+EventLogFile+WHERE+EventType+=+'URI'+
AND+Interval+=+'Hourly' -H "Authorization: Bearer token"
```

In the query, `Interval=Hourly` makes sure that only hourly event log file data is returned. Alternatively, you can use `Sequence` to filter out 24-hour event log files (`Sequence!=0`). To get both hourly and 24-hour files, use `Sequence>=0`.

If your sandbox org has URI events, you see log file records in your query results. You can also download the event log files to review the data in a CSV file. For more information, see Trailhead: Download and Visualize Event Log Files.

Differences Between Hourly and 24-Hour Event Logs

You receive event log files approximately every hour in addition to 24-hour log files. Review the differences between the two logs so that you can filter your files to analyze the event data you want.

Hourly Log Files	24-Hour Log Files
One or more files generated for every hour of activity.	One file generated for every 24 hours of activity.
Available in the API. You can manually import data into third-party visualization apps.	Available in the API and integrated with the Event Monitoring Analytics app and third-party visualization apps.
Key values in the EventLogFile object are: • `Interval`—Hourly • `CreatedDate`—Timestamp of when the log file was created. Use this field to identify new files. • `LogDate`—Timestamp that marks the beginning of the interval when the events occurred. For example, for events that occurred between 11:00 AM and 12:00 PM on 3/7/2016, this field's value is 2016-03-07T11:00:00.000Z.	Key values in the EventLogFile object are: • `Interval`—Daily • `CreatedDate`—Timestamp of when the log file was created. Use this field to identify new files. • `LogDate`—Timestamp that marks the beginning of the interval when the events occurred. For example, for events that occurred on 3/7/2016, this field's value is 2016-03-07T00:00:00.000+0000.

Hourly Log Files	24-Hour Log Files
• `Sequence`—1+. This value increases by 1 when events are added in the same hour after the latest event log file is created. The value resets to 1 in the subsequent hour. See also these Considerations regarding hourly event logs.	• `Sequence`—0
When an hourly incremental log file is delivered, a file with new logs for that hour is created. The `Sequence` field is incremented for each new file.	When a daily incremental log file is delivered, a new file replaces the original file with the full set of available logs for that date. The `CreatedDate` field is updated.

Note: Like with 24-hour event monitoring, hourly event log data is available for the past 30 days.

Using Composite Resources

The examples in this section use composite resources to improve your application's performance by minimizing the number of round-trips between the client and server.

IN THIS SECTION:

Execute Dependent Requests in a Single API Call

The following example uses the Composite resource to execute several dependent requests all in a single call. First, it creates an account and retrieves its information. Next it uses the account data and the Composite resource's reference ID functionality to create a contact and populate its fields based on the account data. Then it retrieves specific information about the account's owner by using query parameters in the request string. Finally, if the metadata has been modified since a certain date, it retrieves account metadata. The `composite.json` file contains the composite request and subrequest data.

Update an Account, Create a Contact, and Link Them with a Junction Object

The following example uses the Composite resource to update some fields on an account, create a contact, and link the two records with a junction object called `AccountContactJunction`. All these requests are executed in a single call. The `composite.json` file contains the composite request and subrequest data.

Update a Record and Get Its Field Values in a Single Request

Use the Composite Batch resource to execute multiple requests in a single API call.

Upsert an Account and Create a Contact

The following example uses the Composite resource to upsert an account and create a contact that is linked to the account. All these requests are executed in a single call. The `composite.json` file contains the composite request and subrequest data.

Create Nested Records

Use the sObject Tree resource to create nested records that share a root record type. For example, in a single request, you can create an account along with its child contacts, and a second account along with its child accounts and contacts. Once the request is processed, the records are created and parents and children are automatically linked by ID. In the request data, you supply the record hierarchies, required and optional field values, each record's type, and a reference ID for each record, and then use the POST method of the resource. The response body will contain the IDs of the created records if the request is successful. Otherwise, the response contains only the reference ID of the record that caused the error and the error information.

Create Multiple Records

While the resource can be used to create nested records, you can also create multiple, unrelated records of the same type. In a single request, you can create up to two hundred records. In the request data, you supply the required and optional field values for each record, each record's type, and a reference ID for each record, and then use the POST method of the resource. The response body will contain the IDs of the created records if the request is successful. Otherwise, the response contains only the reference ID of the record that caused the error and the error information.

Using Composite Graphs

Composite graphs provide an enhanced way to perform composite requests, which execute a series of REST API requests in a single call.

Using a Composite Graph

This example shows how to use a composite graph. It also demonstrates how one request can use more than one composite graph.

allOrNone Parameters in Composite and Collections Requests

If a Composite request uses sObject Collections, there are two or more `allOrNone` parameters that can interact, one on the Composite request and one on each sObject Collections subrequest.

Execute Dependent Requests in a Single API Call

The following example uses the Composite resource to execute several dependent requests all in a single call. First, it creates an account and retrieves its information. Next it uses the account data and the Composite resource's reference ID functionality to create a contact and populate its fields based on the account data. Then it retrieves specific information about the account's owner by using query parameters in the request string. Finally, if the metadata has been modified since a certain date, it retrieves account metadata. The `composite.json` file contains the composite request and subrequest data.

Execute dependent requests in a single API call

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/composite/ -H
"Authorization: Bearer token -H "Content-Type: application/json" -d "@composite.json"
```

Request body `composite.json` file

```
{
    "allOrNone" : true,
    "compositeRequest" : [{
        "method" : "POST",
        "url" : "/services/data/v60.0/sobjects/Account",
        "referenceId" : "NewAccount",
        "body" : {
            "Name" : "Salesforce",
            "BillingStreet" : "Landmark @ 1 Market Street",
            "BillingCity" : "San Francisco",
            "BillingState" : "California",
             "Industry" : "Technology"
        }
    },{
        "method" : "GET",
        "referenceId" : "NewAccountInfo",
        "url" : "/services/data/v60.0/sobjects/Account/@{NewAccount.id}"
    },{
        "method" : "POST",
        "referenceId" : "NewContact",
        "url" : "/services/data/v60.0/sobjects/Contact",
```

```
        "body" : {
            "lastname" : "John Doe",
            "Title" : "CTO of @{NewAccountInfo.Name}",
            "MailingStreet" : "@{NewAccountInfo.BillingStreet}",
            "MailingCity" : "@{NewAccountInfo.BillingAddress.city}",
            "MailingState" : "@{NewAccountInfo.BillingState}",
            "AccountId" : "@{NewAccountInfo.Id}",
            "Email" : "jdoe@salesforce.com",
            "Phone" : "1234567890"
        }
    },{
        "method" : "GET",
        "referenceId" : "NewAccountOwner",
        "url" :
"/services/data/v60.0/sobjects/User/@{NewAccountInfo.OwnerId}?fields=Name,companyName,Title,City,State"

    },{
        "method" : "GET",
        "referenceId" : "AccountMetadata",
        "url" : "/services/data/v60.0/sobjects/Account/describe",
        "httpHeaders" : {
            "If-Modified-Since" : "Tue, 31 May 2016 18:13:37 GMT"
        }
    }]
}
```

Response body after successfully executing the composite request

```
{
    "compositeResponse" : [{
        "body" : {
            "id" : "001R00000033JNuIAM",
            "success" : true,
            "errors" : [ ]
        },
        "httpHeaders" : {
          "Location" : "/services/data/v60.0/sobjects/Account/001R00000033JNuIAM"
        },
        "httpStatusCode" : 201,
        "referenceId" : "NewAccount"
    },{
        "body" : {
            all the account data
        },
        "httpHeaders" : {
            "ETag" : "\"Jbjuzw7dbhaEG3fd90kJbx6A0ow=\"",
            "Last-Modified" : "Fri, 22 Jul 2016 20:19:37 GMT"
        },
        "httpStatusCode" : 200,
        "referenceId" : "NewAccountInfo"
    },{
        "body" : {
            "id" : "003R00000025REHIA2",
            "success" : true,
            "errors" : [ ]
```

```
        },
        "httpHeaders" : {
            "Location" : "/services/data/v60.0/sobjects/Contact/003R00000025REHIA2"
        },
        "httpStatusCode" : 201,
        "referenceId" : "NewContact"
    },{
        "body" : {
            "attributes" : {
            "type" : "User",
            "url" : "/services/data/v60.0/sobjects/User/005R0000000I90CIAS"
            },
            "Name" : "Jane Doe",
            "CompanyName" : "Salesforce",
            "Title" : Director,
            "City" : "San Francisco",
            "State" : "CA",
            "Id" : "005R0000000I90CIAS"
        },
        "httpHeaders" : { },
        "httpStatusCode" : 200,
        "referenceId" : "NewAccountOwner"
    },{
        "body" : null,
        "httpHeaders" : {
            "ETag" : "\"f2293620\"",
            "Last-Modified" : "Fri, 22 Jul 2016 18:45:56 GMT"
         },
        "httpStatusCode" : 304,
        "referenceId" : "AccountMetadata"
    }]
}
```


Update an Account, Create a Contact, and Link Them with a Junction Object

The following example uses the Composite resource to update some fields on an account, create a contact, and link the two records with a junction object called `AccountContactJunction`. All these requests are executed in a single call. The `composite.json` file contains the composite request and subrequest data.

Update an account, create a contact, and link them with a junction object

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/composite/ -H
"Authorization: Bearer token -H "Content-Type: application/json" -d "@composite.json"
```

Request body `composite.json` file


```
{
    "allOrNone" : true,
    "compositeRequest" : [{
        "method" : "PATCH",
        "url" : "/services/data/v60.0/sobjects/Account/001xx000003DIpcAAG",
        "referenceId" : "UpdatedAccount",
        "body" : {
            "Name" : "Salesforce",
```

```
            "BillingStreet" : "Landmark @ 1 Market Street",
            "BillingCity" : "San Francisco",
            "BillingState" : "California",
            "Industry" : "Technology"
        }
    },{
        "method" : "POST",
        "referenceId" : "NewContact",
        "url" : "/services/data/v60.0/sobjects/Contact/",
        "body" : {
            "lastname" : "John Doe",
            "Phone" : "1234567890"
        }
    },{
        "method" : "POST",
        "referenceId" : "JunctionRecord",
        "url" : "/services/data/v60.0/sobjects/AccountContactJunction__c",
        "body" : {
            "accountId__c" : "001xx000003DIpcAAG",
            "contactId__c" : "@{NewContact.id}"
        }
    }]
}
```

Response body after successfully executing the composite request

```
{
  "compositeResponse" : [{
    "body" : null,
    "httpHeaders" : { },
    "httpStatusCode" : 204,
    "referenceId" : "UpdatedAccount"
  }, {
    "body" : {
      "id" : "003R00000025R22IAE",
      "success" : true,
      "errors" : [ ]
    },
    "httpHeaders" : {
      "Location" : "/services/data/v60.0/sobjects/Contact/003R00000025R22IAE"
    },
    "httpStatusCode" : 201,
    "referenceId" : "NewContact"
  }, {
    "body" : {
      "id" : "a00R0000000iN4gIAE",
      "success" : true,
      "errors" : [ ]
    },
    "httpHeaders" : {
      "Location" :
"/services/data/v60.0/sobjects/AccountContactJunction__c/a00R0000000iN4gIAE"
    },
    "httpStatusCode" : 201,
    "referenceId" : "JunctionRecord"
```

```
  }]
}
```

Update a Record and Get Its Field Values in a Single Request

Use the Composite Batch resource to execute multiple requests in a single API call.

The following example updates the name on an account and gets some of the account's field values in a single request. The `batch.json` file contains the subrequest data.

Update a record and query its name and billing postal code in a single request

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/composite/batch/ -H
"Authorization: Bearer token -H "Content-Type: application/json" -d "@batch.json"
```

Request body batch.json file

```
{
"batchRequests" : [
    {
    "method" : "PATCH",
    "url" : "v60.0/sobjects/account/001D000000K0fXOIAZ",
    "richInput" : {"Name" : "NewName"}
    },{
    "method" : "GET",
    "url" : "v60.0/sobjects/account/001D000000K0fXOIAZ?fields=Name,BillingPostalCode"
    }]
}
```

Response body after successfully executing the subrequests

```
{
   "hasErrors" : false,
   "results" : [{
      "statusCode" : 204,
      "result" : null
      },{
      "statusCode" : 200,
      "result": {
         "attributes" : {
            "type" : "Account",
            "url" : "/services/data/v60.0/sobjects/Account/001D000000K0fXOIAZ"
         },
         "Name" : "NewName",
         "BillingPostalCode" : "94105",
         "Id" : "001D000000K0fXOIAZ"
      }
   }]
}
```

SEE ALSO:

Composite Batch

Upsert an Account and Create a Contact

The following example uses the Composite resource to upsert an account and create a contact that is linked to the account. All these requests are executed in a single call. The `composite.json` file contains the composite request and subrequest data.

Upsert an account and create a contact

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/composite/ -H
"Authorization: Bearer token -H "Content-Type: application/json" -d "@composite.json"
```

Request body `composite.json` file

```
{
    "allOrNone" : true,
    "compositeRequest": [{
        "method": "PATCH",
        "url": "/services/data/v60.0/sobjects/Account/ExternalAcctId__c/ID12345",
        "referenceId": "NewAccount",
        "body": {
            "Name": "Acme"
        }
    },{
        "method" : "POST",
        "url" : "/services/data/v60.0/sobjects/Contact",
        "referenceId" : "newContact",
        "body" : {
            "LastName" : "Harrison",
            "AccountId" : "@{NewAccount.id}"
        }
    }]
}
```

Response body after successfully executing the composite request

```
{
    "compositeResponse" : [{
        "body" : {
            "id" : "0016g00000Wqu1EAAR",
            "success" : true,
            "errors" : [ ],
            "created" : true
        },
        "httpHeaders" : {
            "Location" : "/services/data/v60.0/sobjects/Account/0016g00000Wqu1EAAR"
        },
        "httpStatusCode" : 201,
        "referenceId" : "NewAccount"
    },{
        "body" : {
            "id" : "0036g00000WKnfLAAT",
            "success" : true,
            "errors" : [ ]
        },
        "httpHeaders" : {
            "Location" : "/services/data/v60.0/sobjects/Contact/0036g00000WKnfLAAT"
        },
```

```
        "httpStatusCode" : 201,
        "referenceId" : "newContact"
    }]
}
```

SEE ALSO:

sObject Rows by External ID

Create Nested Records

Use the sObject Tree resource to create nested records that share a root record type. For example, in a single request, you can create an account along with its child contacts, and a second account along with its child accounts and contacts. Once the request is processed, the records are created and parents and children are automatically linked by ID. In the request data, you supply the record hierarchies, required and optional field values, each record's type, and a reference ID for each record, and then use the POST method of the resource. The response body will contain the IDs of the created records if the request is successful. Otherwise, the response contains only the reference ID of the record that caused the error and the error information.

The following example creates two sets of nested records. The first set includes an account and two child contact records. The second set includes an account, one child account record, and one child contact record. The record data is provided in `newrecords.json`.

Example for creating two new accounts and their child records

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/composite/tree/Account/
 -H "Authorization: Bearer token -H "Content-Type: application/json" -d "@newrecords.json"
```

Example request body `newrecords.json` file for creating two new Accounts and their child records

```
{
"records" :[{
    "attributes" : {"type" : "Account", "referenceId" : "ref1"},
    "name" : "SampleAccount1",
    "phone" : "1234567890",
    "website" : "www.salesforce.com",
    "numberOfEmployees" : "100",
    "industry" : "Banking",
    "Contacts" : {
      "records" : [{
         "attributes" : {"type" : "Contact", "referenceId" : "ref2"},
         "lastname" : "Smith",
         "Title" : "President",
         "email" : "sample@salesforce.com"
         },{
         "attributes" : {"type" : "Contact", "referenceId" : "ref3"},
         "lastname" : "Evans",
         "title" : "Vice President",
         "email" : "sample@salesforce.com"
         }]
      }
    },{
    "attributes" : {"type" : "Account", "referenceId" : "ref4"},
    "name" : "SampleAccount2",
    "phone" : "1234567890",
    "website" : "www.salesforce.com",
```

```
    "numberOfEmployees" : "52000",
    "industry" : "Banking",
    "childAccounts" : {
      "records" : [{
        "attributes" : {"type" : "Account", "referenceId" : "ref5"},
        "name" : "SampleChildAccount1",
        "phone" : "1234567890",
        "website" : "www.salesforce.com",
        "numberOfEmployees" : "100",
        "industry" : "Banking"
        }]
      },
    "Contacts" : {
      "records" : [{
        "attributes" : {"type" : "Contact", "referenceId" : "ref6"},
        "lastname" : "Jones",
        "title" : "President",
        "email" : "sample@salesforce.com"
        }]
      }
  }]
}
```

Example response body after successfully creating records and relationships

```
{
    "hasErrors" : false,
    "results" : [{
     "referenceId" : "ref1",
     "id" : "001D000000K0fXOIAZ"
     },{
     "referenceId" : "ref4",
     "id" : "001D000000K0fXPIAZ"
     },{
     "referenceId" : "ref2",
     "id" : "003D000000QV9n2IAD"
     },{
     "referenceId" : "ref3",
     "id" : "003D000000QV9n3IAD"
     },{
     "referenceId" : "ref5",
     "id" : "001D000000K0fXQIAZ"
     },{
     "referenceId" : "ref6",
     "id" : "003D000000QV9n4IAD"
     }]
}
```

Once the request is processed, all six records are created with the parent-child relationships specified in the request.

SEE ALSO:

sObject Tree

Create Multiple Records

While the resource can be used to create nested records, you can also create multiple, unrelated records of the same type. In a single request, you can create up to two hundred records. In the request data, you supply the required and optional field values for each record, each record's type, and a reference ID for each record, and then use the POST method of the resource. The response body will contain the IDs of the created records if the request is successful. Otherwise, the response contains only the reference ID of the record that caused the error and the error information.

The following example creates four new accounts. The record data is provided in `newrecords.json`.

Example for creating four new accounts

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/composite/tree/Account/
 -H "Authorization: Bearer token -H "Content-Type: application/json" -d "@newrecords.json"
```

Example request body `newrecords.json` file for creating four new accounts

```
{
"records" :[{
    "attributes" : {"type" : "Account", "referenceId" : "ref1"},
    "name" : "SampleAccount1",
    "phone" : "1111111111",
    "website" : "www.salesforce.com",
    "numberOfEmployees" : "100",
    "industry" : "Banking"
    },{
    "attributes" : {"type" : "Account", "referenceId" : "ref2"},
    "name" : "SampleAccount2",
    "phone" : "2222222222",
    "website" : "www.salesforce2.com",
    "numberOfEmployees" : "250",
    "industry" : "Banking"
    },{
    "attributes" : {"type" : "Account", "referenceId" : "ref3"},
    "name" : "SampleAccount3",
    "phone" : "3333333333",
    "website" : "www.salesforce3.com",
    "numberOfEmployees" : "52000",
    "industry" : "Banking"
    },{
    "attributes" : {"type" : "Account", "referenceId" : "ref4"},
    "name" : "SampleAccount4",
    "phone" : "4444444444",
    "website" : "www.salesforce4.com",
    "numberOfEmployees" : "2500",
    "industry" : "Banking"
    }]
}
```

Example response body after successfully creating records

```
{
    "hasErrors" : false,
    "results" : [{
     "referenceId" : "ref1",
     "id" : "001D000000K1YFjIAN"
```

```
        },{
        "referenceId" : "ref2",
        "id" : "001D000000K1YFkIAN"
        },{
        "referenceId" : "ref3",
        "id" : "001D000000K1YFlIAN"
        },{
        "referenceId" : "ref4",
        "id" : "001D000000K1YFmIAN"
        }]
}
```

SEE ALSO:

sObject Tree

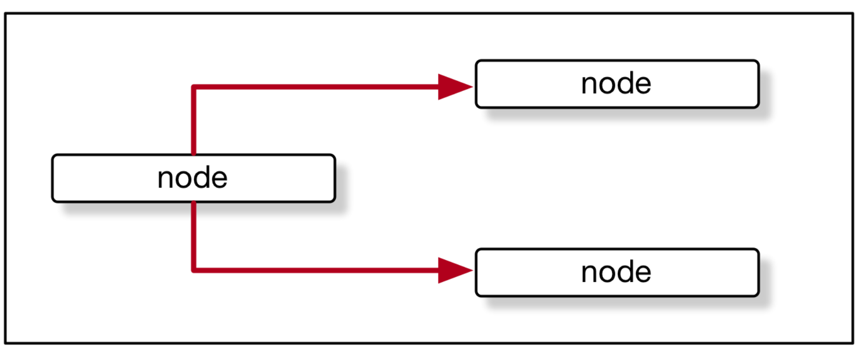
Using Composite Graphs

Composite graphs provide an enhanced way to perform composite requests, which execute a series of REST API requests in a single call.

- Regular composite requests allow you to execute a series of REST API requests in a single call. And you can use the output of one request as the input to a subsequent request.
- Composite graphs extend regular composite requests by allowing you to assemble a more complicated and complete series of related objects and records.
- Composite graphs also enable you to ensure that the steps in a given set of requests are either all completed or all not completed. If you use this option, you don't have to check which requests were successful.
- Regular composite requests have a limit of 25 subrequests. Composite graphs increase this limit to 500.

Defining Composite Graphs

In general, a graph is a collection of connected nodes.



In the context of composite graph operations, the nodes are composite subrequests. For example, a node can be a composite subrequest like this:

```
{
    "url" : "/services/data/v60.0/sobjects/Account/",
    "body" : {
        "name" : "Cloudy Consulting"
```

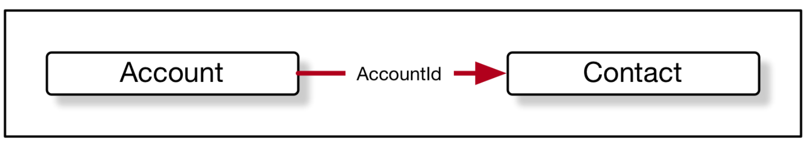
```
    },
    "method" : "POST",
    "referenceId" : "reference_id_account_1"
}
```

Each node features an endpoint representing a record.

Composite graph requests support only these URLs.

URL	Supported HTTP Methods
/services/data/v***XX.X***/sobjects/***sObject***	POST See sObject Basic Information.
/services/data/v***XX.X***/sobjects/***sObject***/***id***	DELETE, GET, PATCH See sObject Rows.
/services/data/v***XX.X***/sobjects/***sObject***/***customFieldName***/***externalId***	DELETE, GET, PATCH, POST See sObject Rows by External ID.

A composite graph can be *directed* meaning that some nodes use information from other nodes. For example, a node that creates a Contact can use the ID of an Account node to link the Contact with the Account.



For example:

```
{
   "graphs": [{
      "graphId": "graph1",
      "compositeRequest": [{
            "body": {
               "name": "Cloudy Consulting"
            },
            "method": "POST",
            "referenceId": "reference_id_account_1",
            "url": "/services/data/v60.0/sobjects/Account/"
         },
         {
            "body": {
               "FirstName": "Nellie",
               "LastName": "Cashman",
               "AccountId": "@{reference_id_account_1.id}"
            },
            "method": "POST",
            "referenceId": "reference_id_contact_1",
            "url": "/services/data/v60.0/sobjects/Contact/"
         }
      ]
```

```
    }]
}
```

Defining Composite Graphs in JSON

A composite graph is defined in JSON like this:

```
{
    "graphId" : "graphId",
    graph
}
```

In other words, like this, where each ***compositeSubrequest*** is a composite subrequest:

```
{
    "graphId" : "graphId",
    "compositeRequest" : [
        compositeSubrequest,
        compositeSubrequest,
        ...
    ]
}
```

The ***graphId*** parameters enable you to identify the graphs when viewing the response. They need not be numeric, but they must follow these restrictions:

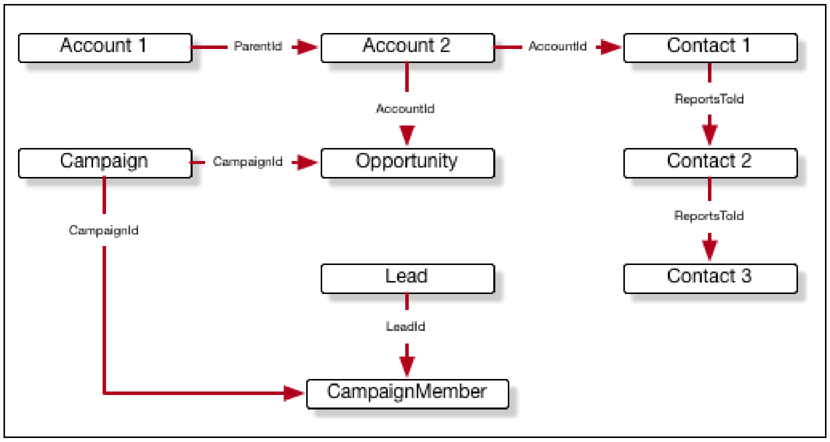
- They must be unique within each composite graph operation.
- They must begin with an alphanumeric character.
- They must be less that 40 characters long.
- They can't contain a period ('.').

A single composite graph request can use one or more composite graphs. See Using a Composite Graph.

Example: Create Accounts, Contacts, Campaigns, Opportunities, Leads, and CampaignMembers with a Composite Graph Request

This example shows a composite graph that performs the following actions:

1. Create Account 1.
2. Create Account 2 as a child of Account 1.
3. Create:
    a. Contact 1, linked to Account 2.
    b. Contact 2, who reports to Contact 1.
    c. Contact 3 who reports to Contact 2.
4. Create a Campaign.
5. Create an Opportunity linked to Account 2 and the Campaign.
6. Create a Lead.
7. Create a CampaignMember linked to the Lead and the Campaign.



The JSON for this graph is:

```
{
   "graphId" : "1",
   "compositeRequest" : [
      {
         "url" : "/services/data/v60.0/sobjects/Account/",
         "body" : {
           "name" : "Cloudy Consulting",
           "description" : "Parent account"
         },
         "method" : "POST",
         "referenceId" : "reference_id_account_1"
      },
      {
         "url" : "/services/data/v60.0/sobjects/Account/",
         "body" : {
           "name" : "Easy Spaces",
           "description" : "Child account",
          ."ParentId" : "@{reference_id_account_1.id}"
         },
         "method" : "POST",
         "referenceId" : "reference_id_account_2"
      },
      {
         "url" : "/services/data/v60.0/sobjects/Contact/",
         "body" : {
           "FirstName" : "Sam",
           "LastName" : "Steele",
           "AccountId" : "@{reference_id_account_2.id}"
         },
         "method" : "POST",
         "referenceId" : "reference_id_contact_1"
      },
      {
         "url" : "/services/data/v60.0/sobjects/Contact/",
         "body" : {
           "FirstName" : "Charlie",
           "LastName" : "Dawson",
           "ReportsToId" : "@{reference_id_contact_1.id}"
```

```
      },
      "method" : "POST",
      "referenceId" : "reference_id_contact_2"
    },
    {
      "url" : "/services/data/v60.0/sobjects/Contact/",
      "body" : {
        "FirstName" : "Nellie",
        "LastName" : "Cashman",
        "ReportsToId" : "@{reference_id_contact_2.id}"
      },
      "method" : "POST",
      "referenceId" : "reference_id_contact_3"
    },
    {
      "url" : "/services/data/v60.0/sobjects/Campaign/",
      "body" : {
        "name" : "Spring Campaign"
      },
      "method" : "POST",
      "referenceId" : "reference_id_campaign"
    },
    {
      "url" : "/services/data/v60.0/sobjects/Opportunity/",
      "body" : {
        "name" : "Opportunity",
        "stageName" : "Value Proposition",
        "closeDate" : "2025-12-31",
        "CampaignId" : "@{reference_id_campaign.id}",
        "AccountId" : "@{reference_id_account_2.id}"
      },
      "method" : "POST",
      "referenceId" : "reference_id_opportunity"
    },
    {
      "url" : "/services/data/v60.0/sobjects/Lead/",
      "body" : {
        "FirstName" : "Belinda",
        "LastName" : "Mulroney",
        "Company" : "Klondike Quarry",
        "Email" : "bmulroney@example.com"
      },
      "method" : "POST",
      "referenceId" : "reference_id_lead"
    },
    {
      "url" : "/services/data/v60.0/sobjects/CampaignMember/",
      "body" : {
        "CampaignId" : "@{reference_id_campaign.id}",
        "LeadId" : "@{reference_id_lead.id}"
      },
      "method" : "POST",
      "referenceId" : "reference_id_campaignmember"
    }
```

```
    ]
}
```

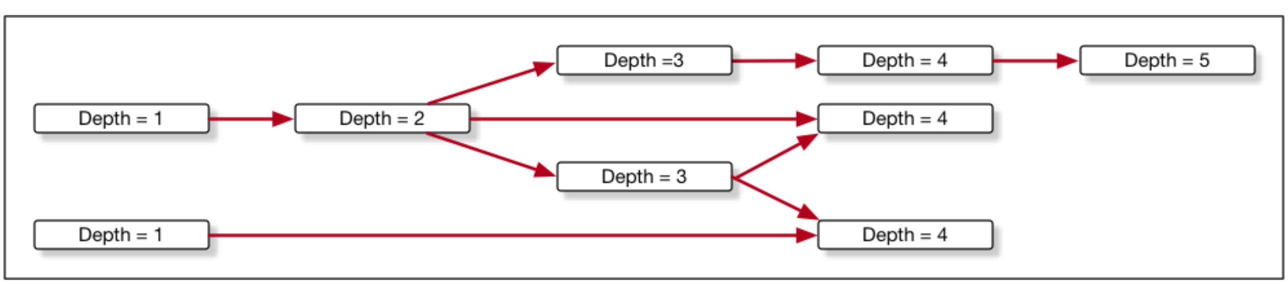
Example: GET Details About a Resource and Then Use Them in a Composite Graph Request

This example shows how you can use GET on a resource, and then use properties of that resource in subsequent requests.

```
{
   "graphs" : [
      {
         "graphId" : "graph1",
         "compositeRequest" : [
            {
               "method" : "GET",
               "url" : "/services/data/v60.0/sobjects/Account/001R0000003fSRrIAM",
               "referenceId" : "refAccount"
            },
            {
               "body" : {
                  "name" : "Amazing opportunity for @{refAccount.Name}",
                  "StageName" : "Stage 1",
                  "CloseDate" : "2025-06-01T23:28:56.782Z",
                  "AccountId" : "@{refAccount.Id}"
               },
               "method" : "POST",
               "url" : "/services/data/v60.0/sobjects/Opportunity",
               "referenceId" : "newOpportunity"
            }
         ]
      }
   ]
}
```

Graph Depth

- Nodes with no parents are considered to be at depth 1.
- The depth of another node is the maximum number of edges in the graph from depth 1 to that node. An edge between two nodes occurs when the one node uses a property (such as `@{reference_account.id}`) from another node.



The AllOrNone Parameter

In standard composite operations, the only control over what happens if an error occurs is the `allOrNone` parameter. If the value is `true`, the entire composite request is rolled back. If the value is `false`, the remaining subrequests that don't depend on the failed subrequest are executed. Dependent subrequests aren't executed, which can lead to a mix of processed and unprocessed records.

Composite graphs have the advantage that each graph is guaranteed to either complete all its subrequests successfully or to be rolled back completely. In other words, the `allOrNone` parameter is implicitly considered to be `true` for each graph. A composite graph request never results in a mix of processed and unprocessed records.

To ensure that graphs are independent, the following rules apply.

1. Subrequests in one graph can't reference subrequests from another graph.
2. Each graph in one composite graph operation must be independent. If one graph can't be processed successfully, that must not prevent other graphs in the same operation from being processed.

Best Practices

As a general practice, keep your graphs as small as possible. For example, it's more efficient to have 50 graphs with 10 nodes rather than 1 graph with 500 nodes. Keeping graphs small has two advantages:

- If one item in a graph fails, only the items in that graph are rolled back. It's easier to identify and handle errors in smaller graphs.
- The server can process multiple, smaller graphs faster and more efficiently.

Example: Submitting a Composite Graph Job

For an example showing how to submit a composite graph job, see Using a Composite Graph.

Using a Composite Graph

This example shows how to use a composite graph. It also demonstrates how one request can use more than one composite graph.

- Regular composite requests allow you to execute a series of REST API requests in a single call. And you can use the output of one request as the input to a subsequent request.
- Composite graphs extend regular composite requests by allowing you to assemble a more complicated and complete series of related objects and records.
- Composite graphs also enable you to ensure that the steps in a given set of requests are either all completed or all not completed. If you use this option, you don't have to check which requests were successful.
- Regular composite requests have a limit of 25 subrequests. Composite graphs increase this limit to 500.

Create a request:

```
curl -X POST curl https://MyDomainName.my.salesforce.com/services/data/v60.0/composite/graph
 -H "Authorization: Bearer token" -H "Content-Type: application/json" --data @data.json
```

where the file `data.json` contains the definition of the graphs. The general format for the request body is:

```
{
  "graphs": [
    {
      "graphId": "graphId",
      "compositeRequest": [
```

```
        compositeSubrequest,
        compositeSubrequest,
        ...
      ]
    },
    {
      "graphId": "graphId",
      "compositeRequest": [
        compositeSubrequest,
        compositeSubrequest,
        ...
      ]
    },
    ...
  ]
}
```

where ***compositeSubrequest*** is a composite subrequest result on page 366.

For example, two composite graph requests each create an Account and then create related records:

```
{
    "graphs" : [
        {
            "graphId" : "1",
            "compositeRequest" : [
                {
                    "url" : "/services/data/v60.0/sobjects/Account/",
                    "body" : {
                        "name" : "Cloudy Consulting"
                    },
                    "method" : "POST",
                    "referenceId" : "reference_id_account_1"
                },
                {
                    "url" : "/services/data/v60.0/sobjects/Contact/",
                    "body" : {
                        "FirstName" : "Nellie",
                        "LastName" : "Cashman",
                        "AccountId" : "@{reference_id_account_1.id}"
                    },
                    "method" : "POST",
                    "referenceId" : "reference_id_contact_1"
                },
                {
                    "url" : "/services/data/v60.0/sobjects/Opportunity/",
                    "body" : {
                        "CloseDate" : "2024-05-22",
                        "StageName" : "Prospecting",
                        "Name" : "Opportunity 1",
                        "AccountId" : "@{reference_id_account_1.id}"
                    },
                    "method" : "POST",
                    "referenceId" : "reference_id_opportunity_1"
                }
```

```
            ]
        },
        {
            "graphId" : "2",
            "compositeRequest" : [
                {
                    "url" : "/services/data/v60.0/sobjects/Account/",
                    "body" : {
                        "name" : "Easy Spaces"
                    },
                    "method" : "POST",
                    "referenceId" : "reference_id_account_2"
                },
                {
                    "url" : "/services/data/v60.0/sobjects/Contact/",
                    "body" : {
                        "FirstName" : "Charlie",
                        "LastName" : "Dawson",
                        "AccountId" : "@{reference_id_account_2.id}"
                    },
                    "method" : "POST",
                    "referenceId" : "reference_id_contact_2"
                }
            ]
        }
    ]
}
```

The response is:

```
{
   "graphs" : [
      {
         "graphId" : "1",
         "graphResponse" : {
            "compositeResponse" : [
               {
                  "body" : {
                     "id" : "001R00000064wc7IAA",
                     "success" : true,
                     "errors" : [ ]
                  },
                  "httpHeaders" : {
                  "Location" : "/services/data/v60.0/sobjects/Account/001R00000064wc7IAA"

                  },
                  "httpStatusCode" : 201,
                  "referenceId" : "reference_id_account_1"
               },
               {
                  "body" : {
                     "id" : "003R000000DDMlTIAX",
                     "success" : true,
                     "errors" : [ ]
                  },
```

```
                  "httpHeaders" : {
                  "Location" : "/services/data/v60.0/sobjects/Contact/003R000000DDMlTIAX"

                  },
                  "httpStatusCode" : 201,
                  "referenceId" : "reference_id_contact_1"
               },
               {
                  "body" : {
                     "id" : "006R0000003FPYxIAO",
                     "success" : true,
                     "errors" : [ ]
                  },
                  "httpHeaders" : {
                     "Location" :
"/services/data/v60.0/sobjects/Opportunity/006R0000003FPYxIAO"
                  },
                  "httpStatusCode" : 201,
                  "referenceId" : "reference_id_opportunity_1"
               }
            ]
         },
         "isSuccessful" : true
      },
      {
         "graphId" : "2",
         "graphResponse" : {
            "compositeResponse" : [
               {
                  "body" : {
                     "id" : "001R00000064wc8IAA",
                     "success" : true,
                     "errors" : [ ]
                  },
                  "httpHeaders" : {
                  "Location" : "/services/data/v60.0/sobjects/Account/001R00000064wc8IAA"

                  },
                  "httpStatusCode" : 201,
                  "referenceId" : "reference_id_account_2"
               },
               {
                  "body" : {
                     "id" : "003R000000DDMlUIAX",
                     "success" : true,
                     "errors" : [ ]
                  },
                  "httpHeaders" : {
                  "Location" : "/services/data/v60.0/sobjects/Contact/003R000000DDMlUIAX"

                  },
                  "httpStatusCode" : 201,
                  "referenceId" : "reference_id_contact_2"
               }
```

```
            ]
         },
         "isSuccessful" : true
      }
   ]
}
```

For more information, see Using Composite Graphs.

`allOrNone` Parameters in Composite and Collections Requests

If a Composite request uses sObject Collections, there are two or more `allOrNone` parameters that can interact, one on the Composite request and one on each sObject Collections subrequest.

- If the Composite request has `allOrNone` set to `true`, then the all-or-none behavior also applies to each of the sObject Collections subrequests, overriding the value of `allOrNone` in the subrequests.
- If the Composite request has `allOrNone` set to `false`, then each sObject Collections subrequest behaves according to its value of `allOrNone`.

Consider, for example, what happens with this job where a Composite request includes two items: an sObject Collections request and a request to create a Contact. The sObject Collections request tries to create two Accounts, one of which fails because there is already an existing Account with the same information.

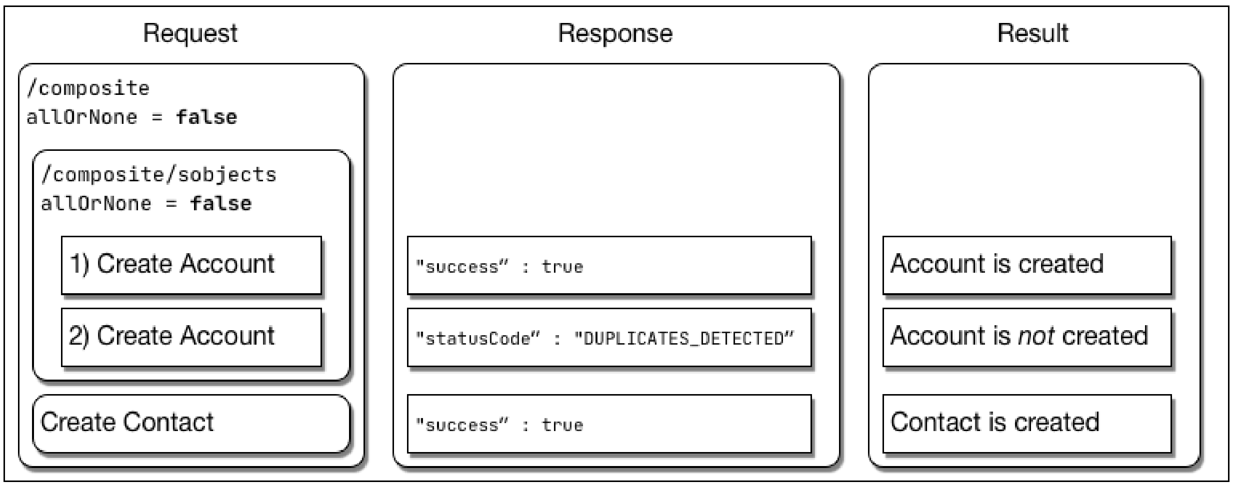
```
POST https://MyDomainName.my.salesforce.com/services/data/v60.0/composite -H "Authorization:
 Bearer token"

{
   "allOrNone" : outerFlag,
   "collateSubrequests" : false,
   "compositeRequest" : [
      {
         "method" : "POST",
         "url" : "/services/data/v60.0/composite/sobjects",
         "body" : {
            "allOrNone" : innerFlag,
            "records" : [
               {
                  "attributes" : { "type" : "Account" },
                  "Name" : "Northern Trail Outfitters",
                  "BillingCity" : "San Francisco"
               },
               {
                  "attributes" : { "type" : "Account" },
                  "Name" : "Easy Spaces",
                  "BillingCity" : "Calgary"
               }
            ]
         },
         "referenceId" : "newAccounts"
      },
      {
         "method" : "POST",
         "url" : "/services/data/v60.0/sObject/Contact",
         "body" : {
```

```
                "FirstName" : "John",
                "LastName" : "Smith"
            },
            "referenceId" : "newContact"
        }
    ]
}
```

The ***outerFlag*** and ***innerFlag*** parameters are either `true` or `false`, which leads to four possible cases.

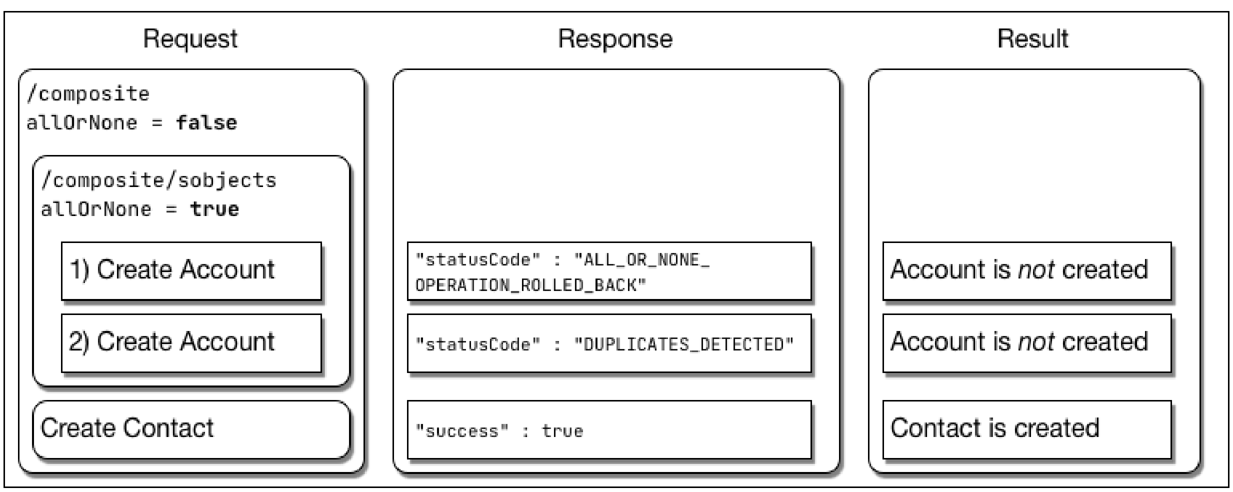
Case 1: *outerFlag* = `false`, *innerFlag* = `false`

In this case, neither request has `allOrNone` set to `true`, so neither request is rolled back. One account is created and one fails.



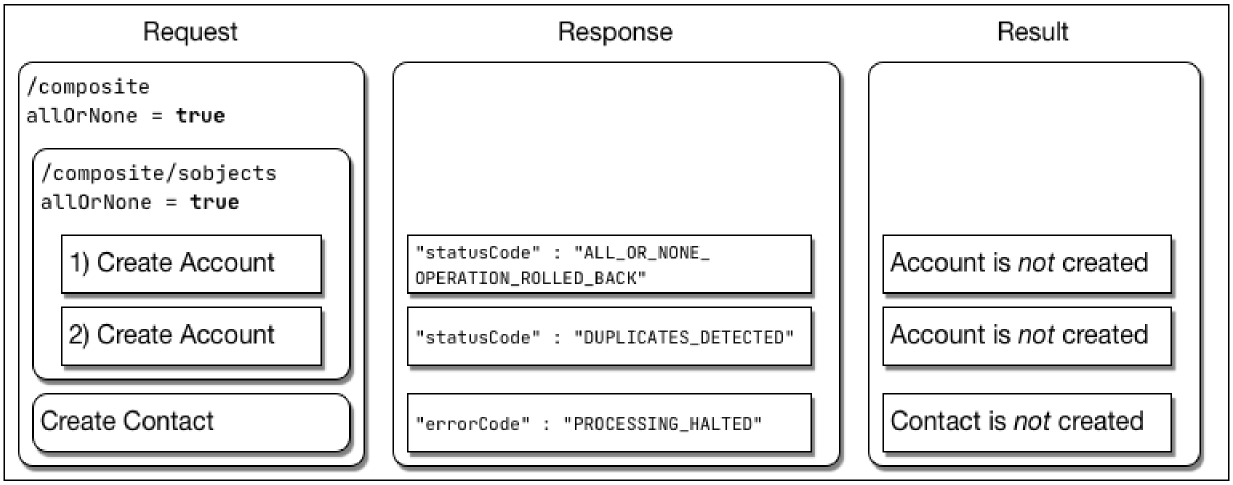
Case 2: *outerFlag* = `false`, *innerFlag* = `true`

In this case, the inner sObject Collections request has `allOrNone` set to `true`, so it is rolled back. The outer Composite request is not rolled back.



Case 3: *outerFlag* = `true`, *innerFlag* = `true`

In this case, both requests have `allOrNone` set to `true`, so they are both rolled back.



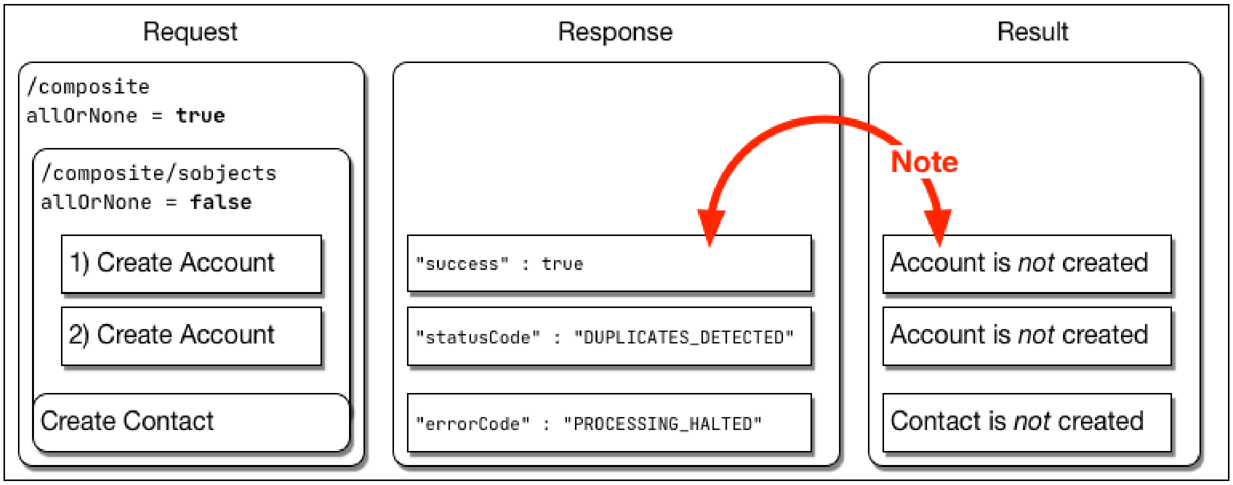
Case 4: *outerFlag* = `true`, ***innerFlag*** = `false`

The response body for this case is:


```
{
   "compositeResponse" : [
      {
         "body" : [
            {
               "id" : "001R00000066cndIAA",
               "success" : true,
               "errors" : [ ]
            },
            {
               "success" : false,
               "errors" : [
                  {
                     "statusCode" : "DUPLICATES_DETECTED",
                     "message" : "Use one of these records?",
                     "fields" : [ ]
                  }
               ]
            }
         ],
         "httpHeaders" : { },
         "httpStatusCode" : 200,
         "referenceId" : "collection1"
      },
      {
         "body" : [
            {
               "errorCode" : "PROCESSING_HALTED",
              "message" : "The transaction was rolled back since another operation in the
 same transaction failed."
            }
         ],
         "httpHeaders" : { },
         "httpStatusCode" : 400,
         "referenceId" : "newContact"
```

```
        }
    ]
}
```

In this case, the inner sObject Collections request has `allOrNone` set to `false`, so it is not immediately rolled back. However, the outer Composite request has `allOrNone` set to `true` so it is rolled back, which also rolls back the inner sObject Collections request.



Note: Even though the response body for sObject Collections request shows `"success" : true` for the creation of the first Account, the fact that the Composite request is rolled back means that the Account creation is rolled back. The final result is that the new Account is *not* created.

CHAPTER 4 Generating an OpenAPI 3.0 Document for sObjects REST API (Beta)

In this Beta, you can generate an OpenAPI document for the sObjects REST API.

Note: This feature is a Beta Service. Customer may opt to try such Beta Service in its sole discretion. Any use of the Beta Service is subject to the applicable Beta Services Terms provided at Agreements and Terms.

With this beta feature, you can generate an OpenAPI document that represents six sObjects REST API resources that reflect your customizations and configurations.

Supported Editions

This beta feature is available to Developer Editions, Partner Developer Editions, sandboxes, and scratch orgs that have API Enabled.

About the Document

The OpenAPI document adheres to Version 3.0.1 of the OpenAPI Specification. For more information, see https://www.openapis.org.

Note: This beta OpenAPI document is subject to change. Don't build production integrations based on this OpenAPI document.

What the OpenAPI Document Covers

This OpenAPI document describes these REST API resources that you can use to retrieve, create, and update your object data.

1. `/services/data/v60.0/sobjects`
   - Lists the available objects and their metadata for your data. Provides the organization encoding, and the maximum batch size permitted in queries.
   - See Describe Global.
2. `/services/data/v60.0/sobjects/`***`sObject`***
   - Describes the individual metadata for the specified object. Can create an object record. For example, retrieve the metadata for the Account object using the GET method or create an Account object using the POST method.
   - See sObject Basic Information.
3. `/services/data/v60.0/sobjects/{sObject}/describe`
   - Describes the individual metadata at all levels for sObjects
   - See sObject Describe.

4. `/services/data/v60.0/sobjects/`***`sObject`***`/{id}`
   - Accesses records based on the specified object ID. Retrieves, updates, or deletes records. Can retrieve field values. Use the GET method to retrieve records or fields, the DELETE method to delete records, and the PATCH method to update records.
   - See sObject Rows.
5. `/services/data/v60.0/sobjects/{sObject}/deleted`
   - Retrieves the list of individual deleted records within the timespan for sObjects
   - See sObject Get Deleted.
6. `/services/data/v`***`XX.X`***`/sobjects/{sObject}/{id}/{blobField}`
   - Retrieves the specified blob field from an individual record and returns it as binary data
   - See sObject Blob Get.

Enabling the Beta

To enable this beta, follow these steps. You must have either the Modify All Data or the Customize Application permission.

1. From Setup, in the Quick Find box, enter *`User Interface`*, and then select **User Interface**.
2. On the User Interface page, select **Enable Salesforce Platform REST API, OpenAPI 3.0 Spec Generation (Beta)**.

   Note: Selecting this item asserts that you accept the Beta Services Terms provided at the Agreements and Terms.

Generating an OpenAPI Document

After the beta feature is enabled, generate the OpenAPI document by following these steps.

1. Send a POST request to

```
https://MyDomainName.my.salesforce.com/services/data/vXX.X/async/specifications/oas3
```

   Note: The procedure in API version 56.0 has changed from the procedure in earlier API versions. This step now uses a POST request instead of a GET request.

   The API version in this POST request, *`XX.X`*, must be 54.0 or greater.

   If the server encounters errors processing the request, it reports a “Failed” status and returns a 4xx or 5xx status.

   The request body must be

```
{
   "resources" : [ selectors ]
}
```

   where *`selectors`* can be

   - `"*"` (to get all the resources)

- or, one or more of the following selectors (to only get specific sections of the OpenAPI document)
  - `"/services/data/v60.0/sobjects"`
  - `"/services/data/v60.0/sobjects/`***`sObject`***`"` where *`sObject`* can be the name of any standard or custom object that you have access to in your org
  - `"/services/data/v60.0/sobjects/`***`sObject`***`/{id}"`
  - `"/services/data/v60.0/sobjects/`***`sObject`***`/deleted"`
  - `"/services/data/v60.0/sobjects/{sObject}/deleted"`
  - `"/services/data/v60.0/sobjects/`***`sObject`***`/describe"`
  - `"/services/data/v60.0/sobjects/{sObject}/describe"`
  - `"/services/data/v60.0/sobjects/{sObject}/{id}/{blobField}"`

  Note: `{blobField}`, `{id}`, and `{sObject}` must be entered literally. They aren't variables.

  Note: Don't add trailing slashes at the end of a selector, For example, `"/services/data/v60.0/sobjects/"` retrieves nothing.

  Note: For the `/describe` and `/deleted` selectors, you can use either `{sObject}` literally or an object name. The response body for these requests is the same for all objects.

  Note: The API version in the selectors must be the latest version, v60.0. (This does not need to be the same version as the POST request in Step 1 or the GET request in Step 3.)

  For example

```
{
   "resources" : ["*"]
}
```

  or

```
{
   "resources" : [
      "/services/data/v60.0/sobjects",
      "/services/data/v60.0/sobjects/Contact",
      "/services/data/v60.0/sobjects/Lead",
      "/services/data/v60.0/sobjects/Lead/{id}",
      "/services/data/v60.0/sobjects/{sObject}/deleted",
      "/services/data/v60.0/sobjects/Account/describe",
     "/services/data/v60.0/sobjects/{sObject}/{id}/{blobField}"

   ]
}
```

2. Assuming that the request can be parsed without errors, the server responds with HTTP status code 202 (Accepted) and a JSON response body that contains a URI where you can get the OpenAPI document. For example:

```
HTTP/1.1 202 Accepted
{
     "href" :
```

```
"/v60.0/async/specifications/oas3/NTByUjAwMDAwMDAwMDBh"
}
```

The last part of this URI (`NTByUjAwMDAwMDAwMDBh` in this example) is the *locator ID* for the OpenAPI document.

3. Send a GET request to the same URI with the locator ID appended. For example:

```
https://MyDomainName.my.salesforce.com/services/data/vXX.X/async/
specifications/oas3/NTByUjAwMDAwMDAwMDBh
```

The API version in this GET request, *`XX.X`*, must be 54.0 or greater.

a. If the server isn't finished preparing the OpenAPI document, it responds with a 202 (Accepted) status code and a status message of "Not Started" or "In Progress". For example:

```
HTTP/1.1 202 Accepted
{
   "status" : "In Progress"
}
```

b. If the server has finished, it responds with a 200 (OK) status and returns the OpenAPI document in the response body. For example:

```
HTTP/1.1 200 OK{
  "openapi": "3.0.1",
  "info": {
      "title": "Lightning Platform REST API",
      "description": "REST API provides you with programmatic
access to your data in Salesforce. The flexibility and
scalability of REST API make it an excellent choice for
integrating Salesforce into your applications and for performing
 complex operations on a large scale. You can use REST API tools
 to create, manipulate, and search data in Salesforce by sending
 HTTP requests to endpoints in Salesforce. Depending on where
you send requests, you access and operate on different pieces
of information, called resources. Resources include records,
query results, metadata, and more. ",
      "version": "60.0
  },
  "servers": [
      ...
  ],
  "security": [
      ...
  ],
  "paths": {
      "/sobjects": { ... },
      "/sobjects/Contact": { ... },
      "/sobjects/Lead": { ... },
      "/sobjects/Lead/{id}": { ... },
      "/sobjects/{sObject}/deleted": { ... },
      "/sobjects/{sObject}/describe": { ... },
      "/sobjects/{sObject}/{id}/{blobField}": { ... }
  },
```

```
  "components": {
      ...
  }
}
```

After the OpenAPI document is generated, you can retrieve the OpenAPI document again by using the same locator ID for 48 hours. After 48 hours, using that locator ID results in a 404 (Not Found) error.

A new OpenAPI document can only be generated every 6 hours per user. If you call `/async/specifications/oas3` again within 6 hours of the last generation, even with different request bodies, the API returns the same locator ID.

If you select a resource that doesn't match any of the supported resources, or you select a resource that you do not have permissions to access, the request does not fail but the OpenAPI document will not contain that resource and its path will not be visible in the OpenAPI document.

Giving Feedback

To give us your feedback, log in to Trailhead and then join the OpenAPI Specs for Salesforce REST APIs Trailblazer Community.

Your feedback is valuable and can help us find existing problems and inspire future change. We're looking for general impressions, improvement suggestions, bugs, and feedback about how well this OpenAPI document aligns with your OpenAPI use cases.

CHAPTER 5 Reference

The following table lists supported REST resources in the API and provides a brief description for each.

In each case, the URI for the resource follows the base URI, `https://`***`MyDomainName`***`.my.salesforce.com`.

For example, to retrieve basic information about an Account object in version 60.0 use `https://`***`MyDomainName`***`.my.salesforce.com/services/data/v60.0/sobjects/Account`.

For information about standard and custom objects that you access with sObject resources, see Object Reference for the Salesforce Platform.

Some of these resources are only available if you have the corresponding package installed or have the corresponding feature enabled.

> **Note:** Some parts of request URIs are case-sensitive, such as IDs. Other parts of URIs aren't case-sensitive, such as object and field names. If your requests aren't successful, check that your URI has the right letter cases by comparing the URI to the reference and examples in this guide.

Resource Name	URI and Description
Versions	`/services/data` Lists summary information about each Salesforce version currently available, including the version, label, and a link to each version's root.
Resources by Version	`/services/data/v`***`XX.X`*** Lists available resources for the specified API version, including resource name and URI.

Invocable Actions	`/services/data/v`***`XX.X`***`/actions/standard`
	`/services/data/v`***`XX.X`***`/actions/custom`
	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.

Analytics (Accesses Reports and Dashboards REST API)	`/services/data/v`***`XX.X`***`/analytics` Accesses Reports and Dashboards REST API resources. See the Salesforce Reports and Dashboards REST API Developer Guide.
Get AppMenu Types	`/services/data/v`***`XX.X`***`/appMenu/` Accesses App Menu types in the Salesforce app dropdown menu.
AppMenu Items	`/services/data/v`***`XX.X`***`/appMenu/AppSwitcher` Accesses App Menu items from the Salesforce app dropdown menu.
AppMenu Mobile Items	`/services/data/v`***`XX.X`***`/appMenu/Salesforce1/`

Resource Name	URI and Description
	Accesses App Menu items from the Salesforce mobile app for Android and iOS and the mobile web navigation menu.
Asset Management	`/services/data/vXX.X/asset-management` Makes make lifecycle-managed asset data available for sales and account reps to view in Lightning Experience. See Customer Asset Lifecycle Management in the *Connect REST API Developer Guide*.
Async Queries	`/services/data/vXX.X/async-queries` Submits a SOQL query to be processed asynchronously. See • Async Query in the *Connect REST API Developer Guide*. • Use Async SOQL with Real-Time Event Monitoring in the *Salesforce Security Guide*. • Async SOQL Use Cases in the *Big Objects Implementation Guide*.
Chatter (Connect REST API)	`/services/data/vXX.X/chatter` Accesses features in the Connect REST API. See the Connect REST API Developer Guide.
Commerce (Place Order REST API)	`/services/data/vXX.X/commerce` Accesses features in the Place Order REST API. See the Place Order REST API Developer Guide.
Compact Layouts	`/services/data/vXX.X/compactLayouts?q=objectList` Returns a list of compact layouts for multiple objects.
Composite	`/services/data/vXX.X/composite` Executes a series of REST API requests in a single POST request, or retrieves a list of other composite resources with a GET request.
Composite Batch	`/services/data/vXX.X/composite/batch` Executes up to 25 subrequests in a single request.
Using Composite Graphs	`/services/data/vXX.X/composite/graph` Provides an enhanced way to perform composite requests.
sObject Tree	`/services/data/vXX.X/composite/tree` Creates one or more sObject trees with root records of the specified type. An sObject tree is a collection of nested, parent-child records with a single root record.
sObject Collections	`/services/data/vXX.X/composite/sobjects` Executes actions on multiple records in one request.
Connect REST API	`/services/data/vXX.X/connect` Accesses features in the Connect REST API. See the Connect REST API Developer Guide.

Resource Name	URI and Description
Financial Services	`/services/data/vXX.X/connect/financialservices` Accesses Financial Services objects. See the Financial Services Cloud Developer Guide.
Health Cloud	`/services/data/vXX.X/connect/health/care-services` Accesses Health Cloud objects. See the Health Cloud Developer Guide.
Manufacturing	`/services/data/vXX.X/connect/manufacturing` Accesses Manufacturing Cloud objects. See the Manufacturing Cloud Developer Guide.

Consumer Goods	`/services/data/vXX.X/connect/object-detection`
	`/services/data/vXX.X/connect/visit/recommendations`
	Accesses the Consumer Goods Business API. See the Consumer Goods Cloud Developer Guide.

Consent	`/services/data/vXX.X/consent` Tracks users' consent preferences.
Contact Tracing	`/services/data/vXX.X/contact-tracing` Tracks health contacts. See the Emergency Response Management for Public Health Developer Guide.
Dedupe	`/services/data/vXX.X/dedupe` Lists duplicate resources, job definitions, and jobs. See the Connect REST API Developer Guide.
Domino	`/services/data/vXX.X/domino` For internal use only.
Eclair (geodata)	`/services/data/vXX.X/eclair` Accesses geodata definitions. See Charts Geodata Resource in the *Analytics REST API Developer Guide*.
Email Connect	`/services/data/vXX.X/emailConnect` For internal use only.
Platform Event Schema by Schema ID	`/services/data/vXX.X/event/eventSchema/schemaId` Gets the definition of a platform event in JSON format for a schema ID. This resource is available in REST API version 40.0 and later.
Folders	`/services/data/vXX.X/folders` Use the Analytics Folders API to perform operations on report and dashboard folders. See Folders in the *Reports and Dashboards REST API Developer Guide*.

Resource Name	URI and Description
Jobs (Bulk API 2.0)	`/services/data/vXX.X/jobs` Accesses jobs in the Bulk API 2.0. See the Bulk API 2.0 and Bulk API Developer Guide.
jsonxform (Analytics REST API)	`/services/data/vXX.X/jsonxform` Tests the results of a rule in a Tableau template. See Rules Testing with jsonxform/transformation endpoint the *Analytics Templates Developer Guide*.
Knowledge Management	`/services/data/vXX.X/knowledgeManagement` Accesses Salesforce Knowledge features. See the Knowledge Developer Guide.
Licensing	`/services/data/vXX.X/licensing` For internal use only.
Limits	`/services/data/vXX.X/limits` Lists information about limits in your org. For each limit, this resource returns the maximum allocation and the remaining allocation based on usage.
Record Count	`/services/data/vXX.X/limits/recordCount` Lists information about object record counts in your organization.
Salesforce Surveys Translation Resources	`/services/data/vXX.X/localizedvalue` Use REST APIs to translate survey fields, view, update, or delete translated survey fields. The translated values of surveys fields are stored in Flow fields.
Metadata	`/services/data/vXX.X/metadata` Accesses features in the Metadata API. See the Metadata API Developer Guide.
Parameterized Search	`/services/data/vXX.X/parameterizedSearch/?q=searchString` Executes a simple REST search using parameters instead of a SOSL clause. Indicate parameters in the URI with the GET method. Or, use the POST method to create complex searches in a request body.
Payments	`/services/data/vXX.X/payments` For internal use only.
Process Approvals	`/services/data/vXX.X/process/approvals` Accesses all approval processes. Can also be used to submit a particular record if that entity supports an approval process and one has already been defined. Records can be approved and rejected if the current user is an assigned approver.
Process Rules	`/services/data/vXX.X/process/rules`

Resource Name	URI and Description
	Accesses a list of all active workflow rules. Use the GET method to retrieve records or fields. Use the HEAD method to retrieve information in HTTP headers. Use the POST method to trigger all active workflow rules.
Query (SOQL)	`/services/data/vXX.X/query/?q=soql` Executes the specified SOQL query.
QueryAll (SOQL)	`/services/data/vXX.X/queryAll/?q=soql` Executes the specified SOQL query. Results can include deleted, merged, and archived records.
Quick Actions	`/services/data/vXX.X/quickActions` Returns a list of global quick actions and their types, as well as custom fields and objects that appear in the Chatter feed.
Recently Viewed Items	`/services/data/vXX.X/recent` Gets the most recently accessed items that were viewed or referenced by the current user.
Salesforce Scheduler Resources	`/services/data/vXX.X/scheduling/` Accesses Scheduler REST APIs to get appointment time slots or available service resources based on work type groups and service territories.
Search (SOSL)	`/services/data/vXX.X/search/?q=sosl` Executes the specified SOSL search. The search string must be URL-encoded.
Search Scope and Order	`/services/data/vXX.X/search/scopeOrder` Returns an ordered list of objects in the default global search scope of a logged-in user. Global search keeps track of which objects the user interacts with and how often and arranges the search results accordingly. Objects used most frequently appear at the top of the list.
Search Suggested Queries	`/services/data/vXX.X/search/suggestSearchQueries?q=searchString&language=languageOfQuery` Returns a list of suggested searches based on the user's query string text matching searches that other users have performed in Salesforce Knowledge. Provides a way to improve search effectiveness, before the user performs a search. This resource is available in REST API version 30.0 and later.
Search Suggested Article Title Matches	`/services/data/vXX.X/search/suggestTitleMatches?q=searchString&language=articleLanguage&publishStatus=articlePublicationStatus` Returns a list of Salesforce Knowledge article titles that match the user's search query string. Provides a shortcut to navigate directly to likely relevant articles before the user performs a search.
Search Result Layouts	`/services/data/vXX.X/searchlayout/?q=commaDelimitedObjectList` Returns search result layout information for the objects in the query string. For each object, this call returns the list of fields displayed on the search results page as columns, the number of rows displayed on the first page, and the label used on the search results page.

Resource Name	URI and Description
Service Templates	`/services/data/vXX.X/serviceTemplates` For internal use only.
Smart Data Discovery (Insights API)	`/services/data/vXX.X/smartdatadiscovery` Uses the Insights API to embed insights into a website, application, or dashboard. See Get Descriptive and Diagnostic Insights Programmatically.
Describe Global	`/services/data/vXX.X/sobjects`
Platform Event Schema by Event Name	`/services/data/vXX.X/sobjects/eventName/eventSchema` Gets the definition of a platform event in JSON format for an event name.
Lightning Exit by Page Metrics	`/services/data/vXX.X/sobjects/LightningExitByPageMetrics` Returns frequency metrics about the standard pages within which users switched from Lightning Experience to Salesforce Classic.
Lightning Toggle Metrics	`/services/data/vXX.X/sobjects/LightningToggleMetrics` Returns details about users who switched between Salesforce Classic and Lightning Experience.
Lightning Usage by App Type	`/services/data/vXX.X/sobjects/LightningUsageByAppTypeMetrics` Returns the total number of Lightning Experience and Salesforce Mobile users. This resource is available in REST API version 44.0 and later.
Lightning Usage by Browser	`/services/data/vXX.X/sobjects/LightningUsageByBrowserMetrics` Returns Lightning Experience usage results grouped by browser instance.
Lightning Usage by FlexiPage	`/services/data/vXX.X/sobjects/LightningUsageByFlexiPageMetrics` Returns details about the custom pages viewed most frequently in Lightning Experience.
Lightning Usage by Page	`/services/data/vXX.X/sobjects/LightningUsageByPageMetrics` Represents standard pages users viewed most frequently in Lightning Experience.
sObject PlatformAction	`/services/data/vXX.X/sobjects/PlatformAction` Queries 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.
sObject Relevant Items	`/services/data/vXX.X/sobjects/relevantItems` Gets the current user's most relevant items. Relevant items include records for objects in the user's global search scope and also most recently used (MRU) objects.
sObject Basic Information	`/services/data/vXX.X/sobjects/sObject` Retrieves basic metadata for a specified object, or creates a new record for the specified object.

Resource Name	URI and Description
sObject Get Deleted	/services/data/v**XX.X**/sobjects/***sObject***/deleted/?start=***startDateAndTime***&end=***endDateAndTime*** Retrieves the list of individual records that have been deleted within the given timespan for the specified object.
sObject Describe	/services/data/v**XX.X**/sobjects/***sObject***/describe Completely describes the individual metadata at all levels for the specified object. For example, this can be used to retrieve the fields, URLs, and child relationships for the Account object.
sObject ApprovalLayouts	/services/data/v**XX.X**/sobjects/***sObject***/describe/approvalLayouts Retrieve a list of approval layouts for a specified object. This resource is available in REST API version 30.0 and later.
sObject CompactLayouts	/services/data/v**XX.X**/sobjects/***sObject***/describe/compactLayouts Retrieve a list of compact layouts for a specific object. This resource is available in REST API version 29.0 and later.
sObject Layouts	/services/data/v**XX.X**/sobjects/***sObject***/describe/layouts /services/data/v**XX.X**/sobjects/Global/describe/layouts Retrieves lists of page layouts and their descriptions. You can request information for all of a specific object's layouts or for layouts associated with a specified record type on a specific object.
sObject Named Layouts	/services/data/v**XX.X**/sobjects/***sObject***/describe/namedLayouts/***layoutName*** Retrieves information about alternate named layouts for a given object. This resource is available in REST API version 31.0 and later.
sObject Rows by External ID	/services/data/v**XX.X**/sobjects/***sObject***/***fieldName***/***fieldValue*** Creates records or updates existing records (upserts records) based on the value of a specified external ID field.
List Views for an Object	/services/data/v**XX.X**/sobjects/***sObject***/listviews /services/data/v**XX.X**/sobjects/***sObject***/listviews/***listViewId*** Returns the list of list views for the specified sObject, including the ID and other basic information about each list view. You can also get basic information for a specific list view by ID.
List View Results	/services/data/vXX.X/sobjects/***sObject***/listviews/***listViewID***/results Executes the SOQL query for the list view and returns the resulting data and presentation information.
List View Describe	/services/data/v**XX.X**/sobjects/***sObject***/listviews/***queryLocator***/describe Returns detailed information about a list view, including the ID, the columns, and the SOQL query.

Resource Name	URI and Description
Recent List Views	`/services/data/v`***XX.X***`/sobjects/`***sObject***`/listviews/recent` Returns the list of recently used list views for the given object.
sObject Quick Actions	`/services/data/v`***XX.X***`/sobjects/`***sObject***`/quickActions/` Access an object's actions and the action details.
sObject Get Updated	`/services/data/v`***XX.X***`/sobjects/`***sObject***`/updated/` `?start=`***startDateAndTime***`&end=`***endDateAndTime***
sObject Rows	`/services/data/v`***XX.X***`/sobjects/`***sObject***`/`***id*** Accesses records based on a specified object and record ID. Retrieves, updates, or deletes records based on the HTTP method. Use the GET method to retrieve records or specific field values, the DELETE method to delete records, or the PATCH method to update records.
sObject Blob Get	`/services/data/v`***XX.X***`/sobjects/`***sObject***`/`***id***`/`***blobField*** Retrieves the specified blob field from an individual record and returns it as binary data.
sObject Relationships	`/services/data/v`***XX.X***`/sobjects/`***sObject***`/`***id***`/`***relationshipName*** Accesses records by traversing object relationships via friendly URLs. You can retrieve, update, or delete the record associated with the traversed relationship field. If there are multiple related records, you can retrieve the complete set of associated records.
sObject Rich Text Image Get	`/services/data/v`***XX.X***`/sobjects/`***sObject***`/`***id***`/` `richTextImageFields/`***fieldName***`/`***contentReferenceId*** Gets the specified image data from a specific rich text area field in a given record.
Delete Lightning Experience Event Series	`/services/data/v`***XX.X***`/sobjects/Event/`***id***`/fromThisEventOnwards` Removes one or more IsRecurrence2 events in a series.
Streaming Channel Push (Steaming API)	`/services/data/vXX.X/sobjects/StreamingChannel/`***channelId***`/push` Gets subscriber information, and pushes notifications for streaming channels. See the Streaming API Developer Guide.
sObject User Password	`/services/data/v`***XX.X***`/sobjects/User/`***userId***`/password` Accesses user passwords based on the specified user ID. Sets, resets, or gets the expiration status of a user password based on the HTTP method. Use the GET method to retrieve a password's expiration status, the POST method to set a password, or the DELETE method to initiate a password reset.
sObject Self-Service User Password	`/services/data/v`***XX.X***`/sobjects/SelfServiceUser/` ***selfServiceUserId***`/password` Accesses self-service user passwords based on the specified user ID. Sets, resets, or gets the expiration status of a self-service user password based on the HTTP method. Use the GET method to retrieve a

Resource Name	URI and Description
	password's expiration status, the POST method to set a password, or the DELETE method to initiate a password reset.
Data Category Groups	`/services/data/vXX.X/support/dataCategoryGroups` Gets data category groups that are visible to the current user.
Data Category Detail	`/services/data/vXX.X/support/dataCategoryGroups/group/dataCategories/category` Gets data category details and the child categories by a given category.
Embedded Service Configuration Describe	`/services/data/vXX.X/support/embeddedservice/configuration/serviceName` Returns information corresponding to one or more service report templates in field service.
Field Service Flow	`/services/data/vXX.X/support/fieldservice/Flow?developerNames=flowName` Returns information corresponding to a field service flow. See Field Service Flow in the *Field Service Developer Guide*.
Field Service Report Template	`/services/data/vXX.X/support/fieldservice/ServiceReportTemplate` Returns information corresponding to one or more service report templates in field service. See Service Report Template in the *Field Service Developer Guide*.

Articles Details	`/services/data/vXX.X/support/knowledgeArticles/articleId`
	`/services/data/vXX.X/support/knowledgeArticles/articleUrlName`
	Gets all online article fields, accessible to the user.

Tabs	`/services/data/vXX.X/tabs` Returns a list of all tabs—including Lightning page tabs—available to the current user, regardless of whether the user has chosen to hide tabs via the All Tabs (+) tab customization feature.
Themes	`/services/data/vXX.X/theme` Gets the list of icons and colors used by themes in the Salesforce application.
Tooling API	`/services/data/vXX.X/tooling` Accesses features in the Tooling API. See the Tooling API Developer Guide.
UI API	`/services/data/vXX.X/ui-api` Accesses features in the UI API. See the User Interface API Developer Guide.
Wave	`/services/data/vXX.X/wave`

Resource Name	URI and Description
(Analytics REST API)	Accesses features in the Analytics REST API. See the Analytics REST API Developer Guide.

Versions

Lists summary information about each Salesforce version currently available, including the version, label, and a link to each version's root.

Syntax

URI

`/services/data/`

Formats

JSON, XML

HTTP Method

GET

Authentication

none

Parameters

none

Example

See List Available REST API Versions on page 31.

Resources by Version

Lists available resources for the specified API version, including resource name and URI.

Syntax

URI

/services/data/v***XX.X***/

Formats

JSON, XML

HTTP Method

GET

Authentication

Authorization: Bearer ***token***

Parameters

none

Example

See List Available REST Resources. on page 36

Limits

Lists information about limits in your org. For each limit, this resource returns the maximum allocation and the remaining allocation based on usage.

This resource is available in REST API version 29.0 and later for API users with the View Setup and Configuration permission.

Syntax

URI

`/services/data/vXX.X/limits/`

Formats

JSON, XML

HTTP Method

GET

Authentication

`Authorization: Bearer token`

Response Body

Limit Label	Description
Analytics	
`AnalyticsExternalDataSizeMB`	Maximum amount of external data in bytes that can be uploaded daily via REST API
`ConcurrentAsyncGetReportInstances`	Concurrent REST API requests for results of asynchronous report runs
`ConcurrentEinsteinDataInsightsStoryCreation`	Concurrent Einstein Discovery data insights story creation via REST API
`ConcurrentEinsteinDiscoveryStoryCreation`	Concurrent Einstein Discovery story creation via REST API
`ConcurrentSyncReportRuns`	Concurrent synchronous report runs via REST API
`HourlyAsyncReportRuns`	Hourly asynchronous report runs via REST API
`HourlySyncReportRuns`	Hourly synchronous report runs via REST API
`HourlyDashboardRefreshes`	Hourly dashboard refreshes via REST API
`HourlyDashboardResults`	Hourly REST API requests for dashboard results
`HourlyDashboardStatuses`	Hourly dashboard status requests via REST API
`DailyAnalyticsDataflowJobExecutions`	Daily analytics dataflow job executions via REST API
`DailyAnalyticsUploadedFilesSizeMB`	Daily cumulative size of analytics files uploaded, in MB
`DailyEinsteinDataInsightsStoryCreation`	Daily Einstein Discovery data insight stores created via REST API

Limit Label	Description
`DailyEinsteinDiscoveryPredictAPICalls`	Daily cumulative number of predicted Einstein Discovery REST API requests
`DailyEinsteinDiscoveryPredictionsByCDC`	Daily cumulative number of predicted Einstein Discovery change data capture add-ons created
`DailyEinsteinDiscoveryOptimizationJobRuns`	Daily cumulative number of Einstein Discovery optimization job runs
`DailyEinsteinDiscoveryStoryCreation`	Daily cumulative number of Einstein Discovery stories created via REST API
`MonthlyEinsteinDiscoveryStoryCreation`	Monthly cumulative number of Einstein Discovery stories created via REST API
Email	
`MassEmail`	Daily number of mass emails that are sent to external email addresses via Apex or APIs
`SingleEmail`	Daily number of single emails that are sent to external email addresses Note: For orgs created before Spring '19, the daily limit is enforced only for emails sent via Apex and Salesforce APIs except for REST API. For orgs created in Spring '19 and later, the daily limit is also enforced for email alerts, simple email actions, Send Email actions in flows, and REST API. If one of the newly counted emails can't be sent because your org has reached the limit, we notify you by email and add an entry to the debug logs.
Lightning Platform REST and Bulk APIs	
`CreateCustom`	Maximum number of allowed custom permission sets. This limit is available for API version 41.0 and later.
`DailyApiRequests`	Daily API calls
`DailyAsyncApexExecutions`	Daily number of asynchronous Apex method executions, which includes batch Apex, future methods, Queueable Apex, and scheduled Apex
`DailyAsyncApexTests`	Daily number of asynchronous Apex test executions. This limit is available in API version 56.0 and later.
`DailyBulkApiBatches` (API version 49.0 and later) or `DailyBulkApiRequests` (API version 48.0 and earlier)	Daily Bulk API and Bulk API 2.0 batches In Bulk API, batches are used by both ingest and query operations. In Bulk API 2.0, batches are used only by ingest operations.
`DailyBulkV2QueryFileStorageMB`	Daily storage for queries in Bulk API 2.0 (measured in MB). This limit is available in API version 47.0 and later.
`DailyBulkV2QueryJobs`	Daily number of query jobs in Bulk API 2.0. This limit is available in API version 47.0 and later.
`PermissionSets`	Maximum number of allowed permission sets. Corresponds to the "Permission sets: maximum (created and added as part of an installed

Limit Label	Description
	managed AppExchange package)" feature allocation. This limit is available in API version 41.0 and later.
Platform Events	
These values apply only to platform events. They don't apply to change data capture events.	
`HourlyPublishedPlatformEvents`	High-volume platform event notifications published per hour
`HourlyPublishedStandardVolumePlatformEvents`	Standard-volume platform event notifications published per hour (You can no longer define standard-volume events. New platform events are high volume by default.)
`DailyStandardVolumePlatformEvents`	Daily standard-volume platform event notifications delivered to CometD clients (You can no longer define standard-volume events. New platform events are high volume by default.)
Platform Events and Change Data Capture	
These values apply to platform events and change data capture events.	
`DailyDeliveredPlatformEvents`	**Org Without Add-On License: Daily Usage and Default Allocation** To get the number of high-volume platform events and change events delivered to CometD and Pub/Sub API clients, `empApi` Lightning components, and event relays in the last 24 hours, use `DailyDeliveredPlatformEvents`. This value doesn't apply to other subscribers, such as Apex triggers, flows, and processes. This value applies to orgs that haven't purchased the high-volume platform event or Change Data Capture add-on. Usage tracking frequency: `DailyDeliveredPlatformEvents` is updated within a few minutes after event delivery.
`MonthlyPlatformEvents` (API version 47.0 and earlier)	**Org With Add-On License: Monthly Event Delivery Usage** If your org purchased the high-volume platform event or Change Data Capture add-on, use MonthlyPlatformEvents. Or to get the usage-based entitlement, see the entry for MonthlyPlatformEventsUsageEntitlement in this table. To get the monthly delivery usage for both high-volume platform events and change events to CometD and Pub/Sub API clients, `empApi` Lightning components, and event relays, use `MonthlyPlatformEvents`. This value doesn't apply to other subscribers, such as Apex triggers, flows, and processes. Usage tracking frequency: `MonthlyPlatformEvents` is updated within a few minutes after event delivery.

Limit Label	Description
MonthlyPlatformEventsUsageEntitlement (API version 48.0 and later)	**Org With Add-On License: Monthly Usage-Based Entitlement** If your org purchased the high-volume platform event or Change Data Capture add-on, use MonthlyPlatformEventsUsageEntitlement. This value is the monthly entitlement and usage of event delivery to CometD and Pub/Sub API clients, empApi Lightning components, and event relays and is incremented for each client. This value doesn't apply to other subscribers, such as Apex triggers, flows, and processes. This value includes usage for both high-volume platform events and change events. Usage tracking frequency: MonthlyPlatformEventsUsageEntitlement is updated once a day. The entitlement is reset every month after your contract start date. Entitlement usage is computed only for production orgs. It's not available in sandbox or trial orgs. For more information, see Usage-based Entitlement Fields. With an add-on license, you can exceed the maximum entitlement by a certain amount. As a result, the remaining value returned can be negative if you exceeded the maximum value. Use MonthlyPlatformEventsUsageEntitlement with API version 48.0 or later to get an accurate event delivery usage based on the first day of your contract. In API version 47.0 and earlier, the MonthlyPlatformEvents value returns the usage based on the first of the month instead of the contract start date.
Private Connect	
PrivateConnectOutboundCalloutHourlyLimitMB	Maximum amount of data in bytes that can be transferred per hour via outbound private connections.
Salesforce Connect	
HourlyLongTermIdMapping	Hourly new long-term external record ID mappings
HourlyODataCallout	Hourly OData callouts
HourlyShortTermIdMapping	Hourly new short-term external record ID mappings
Salesforce Developer Experience	
ActiveScratchOrgs	The maximum number of scratch orgs you can have at any given time based on the edition type. Allocation becomes available if you delete a scratch org or if a scratch org expires.
DailyScratchOrgs	The maximum number of successful scratch org creations you can initiate in a rolling (sliding) 24-hour window. Allocations are determined based on the number of scratch orgs created in the preceding 24 hours.

Limit Label	Description
`Package2VersionCreates`	Daily number of package versions that you can create. Applies to unlocked and second-generation managed packages.
`Package2VersionCreatesWithoutValidation`	Daily number of package versions that skip validation that you can create. Applies to unlocked and second-generation managed packages.
Salesforce Functions	
`DailyFunctionsApiCallLimit` (API version 53.0 and later)	Daily API calls in an org with Functions. Values are visible only if Salesforce Functions is enabled. For more information, see Functions Limits.
Storage	
`DataStorageMB`	Data storage (MB) The API user must have the Manage Users permission.
`FileStorageMB`	File storage (MB) The API user must have the Manage Users permission.
Streaming API—Durable (API version 37.0 and later)	
`DailyDurableGenericStreamingApiEvents`	Generic events notifications delivered in the past 24 hours to all CometD clients
`DailyDurableStreamingApiEvents`	PushTopic event notifications delivered in the past 24 hours to all CometD clients
`DurableStreamingApiConcurrentClients`	Concurrent CometD clients (subscribers) across all channels and for all event types
Streaming API (API version 36.0 and earlier)	
`DailyGenericStreamingApiEvents`	Generic events notifications delivered in the past 24 hours to all CometD clients
`DailyStreamingApiEvents`	PushTopic event notifications delivered in the past 24 hours to all CometD clients
`StreamingApiConcurrentClients`	Concurrent CometD clients (subscribers) across all channels and for all event types
Workflows	
`DailyWorkflowEmails`	Daily workflow emails
`HourlyTimeBasedWorkflow`	Hourly workflow time triggers

Example

See List Org Limits.

Describe Global

Lists the available objects and the associated metadata.

In addition, it provides the organization encoding, as well as the maximum batch size permitted in queries. For more information on encoding, see Internationalization and Character Sets.

You can use the `If-Modified-Since` or `If-Unmodified-Since` header with this resource. When using the `If-Modified-Since` header, if no available object's metadata has changed since the provided date, a `304 Not Modified` status code is returned with no response body.

Note: The `If-Modified-Since` and `If-Unmodified-Since` headers check for more than object-specific metadata changes. They also check for org-wide events, such as changes to permissions, profiles, or field labels.

Syntax

URI

`/services/data/vXX.X/sobjects/`

Formats

JSON, XML

HTTP Method

GET

Authentication

`Authorization: Bearer token`

Parameters

Parameter	Description
`If-Modified-Since`	An optional header specifying a date and time. The request returns records that have been modified after that date and time. The format is `EEE, dd MMM yyyy HH:mm:ss z`. For example: `If-Modified-Since: Mon, 30 Nov 2020 08:34:54 MST`.
`If-Unmodified-Since`	An optional header specifying a date and time. The request returns records that have not been modified after that date and time. The format is `EEE, dd MMM yyyy HH:mm:ss z`. For example: `If-Unmodified-Since: Mon, 30 Nov 2020 08:34:54 MST`.

Example

See Get a List of Objects on page 39.

SEE ALSO:

Conditional Request Headers

sObject Basic Information

Retrieves basic metadata for a specified object, or creates a new record for the specified object.

For example, this resource can be used to retrieve the metadata for the Account object using the GET method, or create a new Account object using the POST method.

IN THIS SECTION:

Get Object Metadata Using sObject Basic Information

Gets basic metadata for a specified object, including some object properties, recent items, and URIs for other resources related to the object.

Create Records Using sObject Basic Information

Creates a new record for a specified object based on field values in the request body.

Get Object Metadata Using sObject Basic Information

Gets basic metadata for a specified object, including some object properties, recent items, and URIs for other resources related to the object.

To retrieve the complete metadata for an object, use the sObject Describe resource.

Syntax

URI

`/services/data/vXX.X/sobjects/sObject/`

Formats

JSON, XML

HTTP Method

GET

Authentication

`Authorization: Bearer token`

Parameters

Parameter	Description
`sObject`	The name of the object. For example, `Account`. A required path parameter.

Example

For an example of retrieving metadata for an object, see Get Metadata for an Object on page 40.

SEE ALSO:

Object Reference for the Salesforce Platform

Create Records Using sObject Basic Information

Creates a new record for a specified object based on field values in the request body.

You must specify values for required fields in the request body. Specifying values for other fields is optional.

Syntax

URI

/services/data/v***XX.X***/sobjects/***sObject***/

Formats

JSON, XML

HTTP Method

POST

Authentication

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

Parameters

Parameter	Description
`Content-Type`	An optional header, specifying the format for the request and response. Possible choices are: • `Content-Type: application/json` • `Content-Type: application/xml`
`sObject`	The name of the object. For example, `Account`. A required path parameter.

Example

- For an example of creating a new record using POST, see Create a Record on page 44.
- For an example of create a new record along with providing blob data for the record, see Insert or Update Blob Data on page 73.

SEE ALSO:

Object Reference for the Salesforce Platform

sObject Describe

Completely describes the individual metadata at all levels for the specified object. For example, this can be used to retrieve the fields, URLs, and child relationships for the Account object.

For more information about the metadata that is retrieved, see DescribesObjectResult in the *SOAP API Developers Guide*.

You can use the `If-Modified-Since` or `If-Unmodified-Since` header with this resource. When using the `If-Modified-Since` header, if no available object's metadata has changed since the provided date, a `304 Not Modified` status code is returned with no response body.

Syntax

URI

/services/data/v***XX.X***/sobjects/***sObject***/describe/

Formats

JSON, XML

HTTP Method

GET

Authentication

Authorization: Bearer ***token***

Parameters

Parameter	Description
sObject	The name of the object. For example, Account. A required path parameter.
If-Modified-Since	An optional header specifying a date and time. The request returns records that have been modified after that date and time. The format is EEE, dd MMM yyyy HH:mm:ss z. For example: If-Modified-Since: Mon, 30 Nov 2020 08:34:54 MST.
If-Unmodified-Since	An optional header specifying a date and time. The request returns records that have not been modified after that date and time. The format is EEE, dd MMM yyyy HH:mm:ss z. For example: If-Unmodified-Since: Mon, 30 Nov 2020 08:34:54 MST.

Example

See Get Field and Other Metadata for an Object. For an example that uses the If-Modified-Since HTTP header, see Get Object Metadata Changes.

SEE ALSO:

Object Reference for the Salesforce Platform

Conditional Request Headers

sObject Get Deleted

Retrieves the list of individual records that have been deleted within the given timespan for the specified object. This resource is available in REST API version 29.0 and later.

This resource is commonly used in data replication applications. Note the following considerations:

- Deleted records are written to a delete log which this resource accesses. A background process that runs every two hours purges records that have been in an organization's delete log for more than two hours if the number of records is above a certain limit.

Starting with the oldest records, the process purges delete log entries until the delete log is back below the limit. This is done to protect Salesforce from performance issues related to massive delete logs

- Information on deleted records is returned only if the current session user has access to them.
- Results are returned for no more than 15 days previous to the day the call is executed (or earlier if an administrator has purged the Recycle Bin).

See "Data Replication" in the *SOAP API Developer Guide* for additional details on data replication and data replication limits.

Syntax

URI

`/services/data/vXX.X/sobjects/sObject/deleted/?start=startDateAndTime&end=endDateAndTime`

Formats

JSON, XML

HTTP Method

GET

Authentication

`Authorization: Bearer token`

Parameters

Parameter	Description
`start`	Starting date/time (Coordinated Universal Time (UTC)—not local— timezone) of the timespan for which to retrieve the data. The API ignores the seconds portion of the specified dateTime value (for example, 12:30:15 is interpreted as 12:30:00 UTC). The date and time must be formatted as described in Valid Date and DateTime Formats. The date/time value for `start` must chronologically precede `end`. This parameter should be URL-encoded.
`end`	Ending date/time (Coordinated Universal Time (UTC)—not local— timezone) of the timespan for which to retrieve the data. The API ignores the seconds portion of the specified dateTime value (for example, 12:35:15 is interpreted as 12:35:00 UTC). The date and time must be formatted as described in Valid Date and DateTime Formats. This parameter should be URL-encoded

Response Body

Property	Type	Description
`deletedRecords`	array	Array of deleted records that satisfy the start and end dates specified in the request. Each entry contains the record ID and the date and time the record was deleted in ISO 8601 format, using Coordinated Universal Time (UTC) timezone.
`earliestDateAvailable`	String	ISO 8601 format timestamp (Coordinated Universal Time (UTC)—not local— timezone) of the last physically deleted object.
`latestDateCovered`	String	ISO 8601 format timestamp (Coordinated Universal Time (UTC)—not local— time zone) of the last date covered in the request.

Example

For an example of getting a list of deleted items, see Get a List of Deleted Records Within a Given Timeframe.

SEE ALSO:

Object Reference for the Salesforce Platform

sObject Get Updated

Retrieves the list of individual records that have been updated (added or changed) within the given timespan for the specified object. This resource is available in REST API version 29.0 and later.

This resource is commonly used in data replication applications. Note the following considerations:

- Results are returned for no more than 30 days previous to the day the call is executed.
- Your client application can replicate any objects to which it has sufficient permissions. For example, to replicate all data for your organization, your client application must be logged in with "View All Data" access rights to the specified object. Similarly, the objects must be within your sharing rules.
- There is a limit of 600,000 IDs returned from this resource. If more than 600,000 IDs would be returned, EXCEEDED_ID_LIMIT is returned. You can correct the error by choosing start and end dates that are closer together.

See "Data Replication" in the *SOAP API Developer Guide* for additional details on data replication and data replication limits.

Syntax

URI

`/services/data/vXX.X/sobjects/sObject/updated/?start=startDateAndTime&end=endDateAndTime`

Formats

JSON, XML

HTTP Method

GET

Authentication

`Authorization: Bearer token`

Parameters

Parameter	Description
`start`	Starting date/time (Coordinated Universal Time (UTC) time zone—not local— timezone) of the timespan for which to retrieve the data. The API ignores the seconds portion of the specified dateTime value (for example, 12:30:15 is interpreted as 12:30:00 UTC). The date and time must be formatted as described in Valid Date and DateTime Formats. The date/time value for `start` must chronologically precede `end`. This parameter should be URL-encoded
`end`	Ending date/time (Coordinated Universal Time (UTC) time zone—not local— timezone) of the timespan for which to retrieve the data. The API ignores the seconds portion of the specified dateTime value (for example, 12:35:15 is interpreted as 12:35:00 UTC).

Parameter	Description
	The date and time must be formatted as described in Valid Date and DateTime Formats. This parameter should be URL-encoded

Response format

Property	Type	Description
`ids`	array	Array of updated records that satisfy the start and end dates specified in the request. Each entry contains the record ID.
`latestDateCovered`	String	ISO 8601 format timestamp (Coordinated Universal Time (UTC)—not local—time zone) of the last date covered in the request.

Examples

For an example of getting a list of updated deleted items, see Get a List of Updated Records Within a Given Timeframe.

SEE ALSO:

Object Reference for the Salesforce Platform

sObject Named Layouts

Retrieves information about alternate named layouts for a given object. This resource is available in REST API version 31.0 and later.

Use this resource to get information on a named layout for a given object. You must provide a valid named layout name as part of the resource URI.

To get a list of named layouts for a given object, use the sObject Describe resource and look for the "namedLayoutInfos" field in the response body.

Syntax

URI

`/services/data/vXX.X/sobjects/sObject/describe/namedLayouts/layoutName`

Formats

JSON, XML

HTTP Method

GET

Authentication

`Authorization: Bearer token`

Request body

None

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/User/describe/namedLayouts/UserAlt
 -H "Authorization: Bearer token"
```

SEE ALSO:

Object Reference for the Salesforce Platform

sObject Rows

Accesses records based on a specified object and record ID. Retrieves, updates, or deletes records based on the HTTP method. Use the GET method to retrieve records or specific field values, the DELETE method to delete records, or the PATCH method to update records.

To create new records, use the sObject Basic Information or sObject Rows by External ID resources.

IN THIS SECTION:

Get Records Using sObject Rows

Gets a record based on the specified object and record ID. The fields and field values of the record are returned in the response body. This resource can be used with external objects in API version 32.0 and later.

Update Records Using sObject Rows

Updates a record based on the specified object and record ID. Field values provided in the request body replace the existing values in the record. This resource can be used with external objects in API version 32.0 and later.

Delete Records Using sObject Rows

Deletes records based on the specified object and record ID. This resource can be used with external objects in API version 32.0 and later.

Get Records Using sObject Rows

Gets a record based on the specified object and record ID. The fields and field values of the record are returned in the response body. This resource can be used with external objects in API version 32.0 and later.

External objects that are associated with non-high-data-volume external data sources use the 18-character Salesforce ID for the `id`. Otherwise, external objects use the External ID standard field of the external object for the `id`.

For information about the items in the response body, see DescribeSObjectResult in the *SOAP API Developer's Guide*.

If the object is an Account object, the response also contains an ETag header. For example: `ETag: "ddpAdaTHz+GcV35e7NLJ9iKD3XXVqAzXT1Sl2ykkP7g=--gzip"` This ETag can be used with the `If-Match` and `If-None-Match` headers. For more information, see Conditional Request Headers.

Syntax

URI

`/services/data/vXX.X/sobjects/sObject/id/`

Formats

JSON, XML

HTTP Method

GET

Authentication

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

Parameters

Parameter	Description
`sObject`	The name of the object. For example, `Account`.
`id`	The identifier of the object. For example, `001R0000005hDFYIA2`.
`fields`	A comma-delimited list of fields specifying the fields and values returned in the response body. For example, `?fields=name,description,numberofemployees,industry`.
`If-Match`	An optional header specifying a comma-delimited list of one or more ETags. This only has an effect when used with Account objects. The request is only processed if the Account's ETag matches one of the ETags in the list. For example: `If-Match: "94C83JSreaVMGpL+lUzv8Dr3inI0kCvuKATVJcTuApA=--gzip", "ddpAdaTHz+GcV35e7NLJ9iKD3XXVqAzXT1Sl2ykkP7g=--gzip"`.
`If-None-Match`	An optional header specifying a comma-delimited list of one or more ETags. This only has an effect when used with Account objects. The request is only processed if the Account's ETag does not match one of the ETags in the list. For example: `If-None-Match: "94C83JSreaVMGpL+lUzv8Dr3inI0kCvuKATVJcTuApA=--gzip", "ddpAdaTHz+GcV35e7NLJ9iKD3XXVqAzXT1Sl2ykkP7g=--gzip"`.
`If-Modified-Since`	An optional header specifying a date and time. The request returns records that have been modified after that date and time. The format is `EEE, dd MMM yyyy HH:mm:ss z` For example: `If-Modified-Since: Mon, 30 Nov 2020 08:34:54 MST`.
`If-Unmodified-Since`	An optional header specifying a date and time. The request returns records that have not been modified after that date and time. The format is `EEE, dd MMM yyyy HH:mm:ss z` For example: `If-Unmodified-Since: Mon, 30 Nov 2020 08:34:54 MST`.

Example

For examples of retrieving records, see Get Field Values from a Standard Object Record.

SEE ALSO:

Object Reference for the Salesforce Platform

Update Records Using sObject Rows

Updates a record based on the specified object and record ID. Field values provided in the request body replace the existing values in the record. This resource can be used with external objects in API version 32.0 and later.

External objects that are associated with non-high-data-volume external data sources use the 18-character Salesforce ID for the `id`. Otherwise, external objects use the External ID standard field of the external object for the `id`.

For information about the items in the response body, see DescribeSObjectResult in the *SOAP API Developer Guide*.

If the object is an Account object, the response also contains an ETag header. For example: `ETag: "ddpAdaTHz+GcV35e7NLJ9iKD3XXVqAzXT1Sl2ykkP7g=--gzip"` This ETag can be used with the `If-Match` and `If-None-Match` headers. For more information, see Conditional Request Headers.

Syntax

URI

`/services/data/v`***`XX.X`***`/sobjects/`***`sObject`***`/`***`id`***`/`

Formats

JSON, XML

HTTP Method

PATCH

Authentication

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

Parameters

Parameter	Description
`Content-Type`	An optional header that specifies the format for the request and response. The possible header values are: • `Content-Type: application/json` • `Content-Type: application/xml`
`sObject`	The name of the object. For example, `Account` and `CustomObject__c`.
`id`	The identifier of the object. For example, `001R0000005hDFYIA2`.

Parameter	Description
`If-Match`	An optional header specifying a comma-delimited list of one or more ETags. This only has an effect when used with Account objects. The request is only processed if the Account's ETag matches one of the ETags in the list. For example: `If-Match: "94C83JSreaVMGpL+lUzv8Dr3inI0kCvuKATVJcTuApA=--gzip", "ddpAdaTHz+GcV35e7NLJ9iKD3XXVqAzXT1Sl2ykkP7g=--gzip"`.
`If-None-Match`	An optional header specifying a comma-delimited list of one or more ETags. This only has an effect when used with Account objects. The request is only processed if the Account's ETag does not match one of the ETags in the list. For example: `If-None-Match: "94C83JSreaVMGpL+lUzv8Dr3inI0kCvuKATVJcTuApA=--gzip", "ddpAdaTHz+GcV35e7NLJ9iKD3XXVqAzXT1Sl2ykkP7g=--gzip"`.
`If-Modified-Since`	An optional header specifying a date and time. The request returns records that have been modified after that date and time. The format is `EEE, dd MMM yyyy HH:mm:ss z` For example: `If-Modified-Since: Mon, 30 Nov 2020 08:34:54 MST`.
`If-Unmodified-Since`	An optional header specifying a date and time. The request returns records that have not been modified after that date and time. The format is `EEE, dd MMM yyyy HH:mm:ss z` For example: `If-Unmodified-Since: Mon, 30 Nov 2020 08:34:54 MST`.

Example

For an example of updating a record using PATCH, see Update a Record.

SEE ALSO:

Object Reference for the Salesforce Platform

Conditional Request Headers

Delete Records Using sObject Rows

Deletes records based on the specified object and record ID. This resource can be used with external objects in API version 32.0 and later.

External objects that are associated with non-high-data-volume external data sources use the 18-character Salesforce ID for the *id*. Otherwise, external objects use the External ID standard field of the external object for the *id*.

If the object is an Account object, the response also contains an ETag header. For example: `ETag: "ddpAdaTHz+GcV35e7NLJ9iKD3XXVqAzXT1Sl2ykkP7g=--gzip"` This ETag can be used with the `If-Match` and `If-None-Match` headers. For more information, see Conditional Request Headers.

Syntax

URI

`/services/data/vXX.X/sobjects/sObject/id/`

Formats

JSON, XML

HTTP Method

DELETE

Authentication

`Authorization: Bearer token`

Parameters

Parameter	Description
`sObject`	The name of the object. For example, `Account`.
`id`	The identifier of the record. For example, `001R0000005hDFYIA2`.
`If-Match`	An optional header specifying a comma-delimited list of one or more ETags. This only has an effect when used with Account objects. The request is only processed if the Account's ETag matches one of the ETags in the list. For example: `If-Match: "94C83JSreaVMGpL+1Uzv8Dr3inI0kCvuKATVJcTuApA=--gzip", "ddpAdaTHz+GcV35e7NLJ9iKD3XXVqAzXT1Sl2ykkP7g=--gzip"`.
`If-None-Match`	An optional header specifying a comma-delimited list of one or more ETags. This only has an effect when used with Account objects. The request is only processed if the Account's ETag does not match one of the ETags in the list. For example: `If-None-Match: "94C83JSreaVMGpL+1Uzv8Dr3inI0kCvuKATVJcTuApA=--gzip", "ddpAdaTHz+GcV35e7NLJ9iKD3XXVqAzXT1Sl2ykkP7g=--gzip"`.
`If-Modified-Since`	An optional header specifying a date and time. The request returns records that have been modified after that date and time. The format is `EEE, dd MMM yyyy HH:mm:ss z` For example: `If-Modified-Since: Mon, 30 Nov 2020 08:34:54 MST`.
`If-Unmodified-Since`	An optional header specifying a date and time. The request returns records that have not been modified after that date and time. The format is `EEE, dd MMM yyyy HH:mm:ss z`

Parameter	Description
	For example: `If-Unmodified-Since: Mon, 30 Nov 2020 08:34:54 MST`.

Example

For an example of deleting a record using DELETE, see Delete a Record.

SEE ALSO:

Object Reference for the Salesforce Platform

sObject Rows by External ID

Creates, retrieves, upserts, or deletes records based on the value of a specified external ID field. By using the PATCH method with this resource, you can send upsert requests to Salesforce.

IN THIS SECTION:

Get Records Using sObject Rows by External ID
Retrieves a record based on the value of the specified external ID field.

Create Records Using sObject Rows by External ID
Creates a new record based on the field values included in the request body. This resource does not require the use of an external ID field.

Upsert Records Using sObject Rows by External ID
Upserts a record based on the value of the specified external ID field. Based on whether the value of the external ID already exists, the request either creates a new record or updates an existing one.

Delete Records Using sObject Rows by External ID
Deletes a record based on the value of the specified external ID field.

Return Headers Using sObject Rows by External ID
Returns only the headers that are returned by sending a GET request to the sObject Rows by External ID resource. This gives you a chance to see returned header values of the GET request before retrieving the content itself.

Get Records Using sObject Rows by External ID

Retrieves a record based on the value of the specified external ID field.

Note: For security reasons, some Top Level Domains (TLD) can conflict with certain file format extensions. Adjust your implementation to work around such cases.

For example, using an email address, such as `example@email.inc`, as the External ID returns a "404 not found" error.

There are several workarounds to handle conflicting TLDs.

- Use a different External ID field.
- Create a new External ID field that is the same as the email field and replace the "." with an "_" to handle this case.
- Run a query for emails ending in ".inc" to retrieve the record ID and use that for the upsert request.

- Use SOAP API instead of REST API for upsert requests.
- Accept the email as a query parameter instead of a path parameter by creating a custom Apex REST API. Use Apex to perform the upsert request.

Syntax

URI

`/services/data/vXX.X/sobjects/sObject/fieldName/fieldValue`

Formats

JSON, XML

HTTP Method

GET

Authentication

`Authorization: Bearer token`

Parameters

None

Example

For an example of retrieving a record based on an external ID, see Get a Record Using an External ID on page 48.

SEE ALSO:

Object Reference for the Salesforce Platform

Create Records Using sObject Rows by External ID

Creates a new record based on the field values included in the request body. This resource does not require the use of an external ID field.

As a special case, in API version 37.0 and later, you can create a record with a POST request to `/services/data/vXX.X/sobjects/sObjectName/Id`. Because `Id` has a `null` value, it is omitted from the request, and the record is created according to the request body. Creating records with this resource is useful because you can use the same URI in each POST request for each new record. In this case, you are not required to specify an external ID to create a record.

Note: Do not specify `Id` or an external ID field in the request body or an error is generated.

Syntax

URI

`/services/data/vXX.X/sobjects/sObject/Id`

Formats

JSON, XML

HTTP Method

POST

Authentication

`Authorization: Bearer token`

Parameters

None

Example

Example Request

```
curl -X POST
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/Id -H
"Authorization: Bearer token" -H "Content-Type: application/json" -d "@newaccount.json"
```

SEE ALSO:

Object Reference for the Salesforce Platform

Upsert Records Using sObject Rows by External ID

Upserts a record based on the value of the specified external ID field. Based on whether the value of the external ID already exists, the request either creates a new record or updates an existing one.

- If the external ID doesn't match an existing record, then a new record is created according to the request body.
- If the external ID matches one existing record, then the existing record is updated according to the request body.
- If the external ID matches multiple existing records, then a 300 error is returned, and no records are created or updated.

If you are upserting a record for an object that has a custom field with both the `External ID` and `Unique` attributes selected (a unique index), you do not need any special permissions. The `Unique` attribute prevents the creation of duplicates. If you are upserting a record for an object that has the `External ID` attribute selected but not the `Unique` attribute selected, (a non-unique index) your client application must have the permission "View All Data" to execute this call.

Note: For security reasons, some Top Level Domains (TLD) can conflict with certain file format extensions. Adjust your implementation to work around such cases.

For example, using an email address, such as `example@email.inc`, as the External ID returns a "404 not found" error.

There are several workarounds to handle conflicting TLDs.

- Use a different External ID field.
- Create a new External ID field that is the same as the email field and replace the "." with an "_" to handle this case.
- Run a query for emails ending in ".inc" to retrieve the record ID and use that for the upsert request.
- Use SOAP API instead of REST API for upsert requests.
- Accept the email as a query parameter instead of a path parameter by creating a custom Apex REST API. Use Apex to perform the upsert request.

Syntax

URI

`/services/data/vXX.X/sobjects/sObject/fieldName/fieldValue`

Formats

JSON, XML

HTTP Method

PATCH

Authentication

`Authorization: Bearer token`

Parameters

None

Example

For examples of creating and updating records based on external IDs, see Insert or Update (Upsert) a Record Using an External ID on page 48.

SEE ALSO:

Object Reference for the Salesforce Platform

Delete Records Using sObject Rows by External ID

Deletes a record based on the value of the specified external ID field.

Note: For security reasons, some Top Level Domains (TLD) can conflict with certain file format extensions. Adjust your implementation to work around such cases.

For example, using an email address, such as `example@email.inc`, as the External ID returns a “404 not found” error.

There are several workarounds to handle conflicting TLDs.

- Use a different External ID field.
- Create a new External ID field that is the same as the email field and replace the "." with an "_" to handle this case.
- Run a query for emails ending in ".inc" to retrieve the record ID and use that for the upsert request.
- Use SOAP API instead of REST API for upsert requests.
- Accept the email as a query parameter instead of a path parameter by creating a custom Apex REST API. Use Apex to perform the upsert request.

Syntax

URI

`/services/data/vXX.X/sobjects/sObject/fieldName/fieldValue`

Formats

JSON, XML

HTTP Method

DELETE

Authentication

`Authorization: Bearer token`

Parameters

None

SEE ALSO:

Object Reference for the Salesforce Platform

Return Headers Using sObject Rows by External ID

Returns only the headers that are returned by sending a GET request to the sObject Rows by External ID resource. This gives you a chance to see returned header values of the GET request before retrieving the content itself.

Note: For security reasons, some Top Level Domains (TLD) can conflict with certain file format extensions. Adjust your implementation to work around such cases.

For example, using an email address, such as `example@email.inc`, as the External ID returns a “404 not found” error.

There are several workarounds to handle conflicting TLDs.

- Use a different External ID field.
- Create a new External ID field that is the same as the email field and replace the "." with an "_" to handle this case.
- Run a query for emails ending in ".inc" to retrieve the record ID and use that for the upsert request.
- Use SOAP API instead of REST API for upsert requests.
- Accept the email as a query parameter instead of a path parameter by creating a custom Apex REST API. Use Apex to perform the upsert request.

Syntax

URI

`/services/data/vXX.X/sobjects/sObject/fieldName/fieldValue`

Formats

JSON, XML

HTTP Method

HEAD

Authentication

`Authorization: Bearer token`

Parameters

None

SEE ALSO:

Object Reference for the Salesforce Platform

sObject Blob Get

Gets the specified blob field from an individual record and returns it as binary data. Only certain standard objects have blob fields, such as Attachment, ContentNote, ContentVersion, Document, Folder, and Note.

Note: The sObject Blob Get resource isn't compatible with Composite API requests, because it returns binary data instead of data in JSON or XML formats. Instead, make individual sObject Blob Get requests to retrieve blob data.

Syntax

URI

`/services/data/vXX.X/sobjects/sObject/id/blobField`

Formats

Binary data

HTTP Method

GET

Authentication

`Authorization: Bearer token`

Parameters

none required

Example

For an example of retrieving blob data from a Document, see Get Blob Data on page 79.

SEE ALSO:

Object Reference for the Salesforce Platform

sObject ApprovalLayouts

Retrieve a list of approval layouts for a specified object. This resource is available in REST API version 30.0 and later.

IN THIS SECTION:

Get Approval Layouts

Gets a list of approval layouts for a specified object. This resource is available in REST API version 30.0 and later.

Return Headers for Approval Layouts

Returns only the headers that are returned by a GET request to sObject ApprovalLayouts resources. This gives you a chance to see header values before retrieving the content of the resource. This resource is available in REST API version 30.0 and later.

SEE ALSO:

Object Reference for the Salesforce Platform

Get Approval Layouts

Gets a list of approval layouts for a specified object. This resource is available in REST API version 30.0 and later.

Syntax

URI

/services/data/v***XX.X***/sobjects/***sObject***/describe/approvalLayouts/

Formats

JSON, XML

HTTP methods

GET

Authentication

Authorization: Bearer ***token***

Request parameters

None required

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/describe/approvalLayouts/
 -H "Authorization: Bearer token"
```

Example Response Body


```
{
  "approvalLayouts" : [ {
    "id" : "04aD00000008Py9IAE",
    "label" : "MyApprovalProcessName",
    "layoutItems" : [...],
    "name" : "MyApprovalProcessName"
    }, {
    "id" : "04aD00000008Q0KIAU",
    "label" : "Process1",
    "layoutItems" : [...],
    "name" : "Process1"
  } ]
}
```


If you haven't defined any approval layouts for an object, the response is `{"approvalLayouts" : [ ]}`.

Return Headers for Approval Layouts

Returns only the headers that are returned by a GET request to sObject ApprovalLayouts resources. This gives you a chance to see header values before retrieving the content of the resource. This resource is available in REST API version 30.0 and later.

Syntax

URI

To return headers of a request for an approval layout description for a specified object, use /services/data/v***XX.X***/sobjects/***sObject***/describe/approvalLayouts/

Formats
JSON, XML

HTTP methods
HEAD

Authentication
`Authorization: Bearer token`

Request parameters
None required

Example

Example Request

```
curl -X HEAD --head
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/describe/approvalLayouts/
 -H "Authorization: Bearer token"
```

sObject Single Approval Process

Retrieves an approval layout for a named approval process on a specified object. This resource is available in REST API version 30.0 and later.

Get a Layout for a Single Approval Process on a Specified Object

Retrieves an approval layout for a named approval process on a specified object. This resource is available in REST API version 30.0 and later.

Syntax

URI
`/services/data/vXX.X/sobjects/sObject/describe/approvalLayouts/approvalProcessName`

Formats
JSON, XML

HTTP methods
GET

Authentication
`Authorization: Bearer token`

Request parameters
None required

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/describe/approvalLayouts/ExampleApprovalProcessName
 -H "Authorization: Bearer token"
```

Example Response Body

```
{
  "approvalLayouts" : [ {
    "id" : "04aD00000008Py9IAE",
    "label" : "ExampleApprovalProcessName",
    "layoutItems" : [...],
    "name" : "ExampleApprovalProcessName"
  } ]
}
```

Return Headers for a Single Approval Process on a Specified Object

Returns only the headers that are returned by a GET request to sObject ApprovalLayouts resources. This gives you a chance to see header values before retrieving the content of the resource. Specify a particular approval process name to limit the request to one specific approval layout. This resource is available in REST API version 30.0 and later.

Syntax

URI

`/services/data/vXX.X/sobjects/sObject/describe/approvalLayouts/approvalProcessName`

Formats

JSON, XML

HTTP methods

HEAD

Authentication

`Authorization: Bearer token`

Request parameters

None required

Example

Example Request

```
curl -X HEAD --head
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/describe/approvalLayouts/ExampleApprovalProcessName
 -H "Authorization: Bearer token"
```

sObject CompactLayouts

Retrieve a list of compact layouts for a specific object. This resource is available in REST API version 29.0 and later.

IN THIS SECTION:

Get Compact Layouts Using sObject CompactLayouts

Retrieves a list of compact layouts for a specific object. This resource is available in REST API version 29.0 and later.

Return Headers Using sObject CompactLayouts

Returns only the headers that are returned by a GET request to the sObject CompactLayouts resource. This gives you a chance to see header values ahead of time before retrieving the content of the resource. This resource is available in REST API version 29.0 and later.

SEE ALSO:

Object Reference for the Salesforce Platform

Get Compact Layouts Using sObject CompactLayouts

Retrieves a list of compact layouts for a specific object. This resource is available in REST API version 29.0 and later.

Syntax

URI

`/services/data/vXX.X/sobjects/sObject/describe/compactLayouts/`

Formats

JSON, XML

HTTP methods

GET

Authentication

`Authorization: Bearer token`

Request parameters

None required

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/describe/compactLayouts
 -H "Authorization: Bearer token"
```

Example Response Body

This sample JSON response is for compact layouts created on the Account object. In this example, only one custom compact layout was created for Account. The custom compact layout is assigned as the primary compact layout for the object, and it contains two fields: `Account Name` and `Phone`.

```
{
  "compactLayouts" : [ {
    "actions" : [ {
      "custom" : false,
      "icons" : null,
```

```
    "label" : "Call",
    "name" : "CallHighlightAction"
  }, {
    "custom" : false,
    "icons" : null,
    "label" : "Send Email",
    "name" : "EmailHighlightAction"
  }, {
    "custom" : false,
    "icons" : null,
    "label" : "Map",
    "name" : "MapHighlightAction"
  }, {
    "custom" : false,
    "icons" : null,
    "label" : "Read News",
    "name" : "NewsHighlightAction"
  }, {
    "custom" : false,
    "icons" : null,
    "label" : "View Website",
    "name" : "WebsiteHighlightAction"
  } ],
  "fieldItems" : [ {
    "editable" : false,
    "label" : "Account Name",
    "layoutComponents" : [ {
      "components" : [ ],
      "details" : {
        "autoNumber" : false,
        "byteLength" : 765,
        "calculated" : false,
        "calculatedFormula" : null,
        "cascadeDelete" : false,
        "caseSensitive" : false,
        "controllerName" : null,
        "createable" : true,
        "custom" : false,
        "defaultValue" : null,
        "defaultValueFormula" : null,
        "defaultedOnCreate" : false,
        "dependentPicklist" : false,
        "deprecatedAndHidden" : false,
        "digits" : 0,
        "displayLocationInDecimal" : false,
        "externalId" : false,
        "extraTypeInfo" : null,
        "filterable" : true,
        "groupable" : true,
        "htmlFormatted" : false,
        "idLookup" : false,
        "inlineHelpText" : null,
        "label" : "Account Name",
        "length" : 255,
```

```
        "mask" : null,
        "maskType" : null,
        "name" : "Name",
        "nameField" : true,
        "namePointing" : false,
        "nillable" : false,
        "permissionable" : false,
        "picklistValues" : [ ],
        "precision" : 0,
        "queryByDistance" : false,
        "referenceTo" : [ ],
        "relationshipName" : null,
        "relationshipOrder" : null,
        "restrictedDelete" : false,
        "restrictedPicklist" : false,
        "scale" : 0,
        "soapType" : "xsd:string",
        "sortable" : true,
        "type" : "string",
        "unique" : false,
        "updateable" : true,
        "writeRequiresMasterRead" : false
      },
      "displayLines" : 1,
      "tabOrder" : 2,
      "type" : "Field",
      "value" : "Name"
    } ],
    "placeholder" : false,
    "required" : false
  }, {
    "editable" : false,
    "label" : "Phone",
    "layoutComponents" : [ {
      "components" : [ ],
      "details" : {
        "autoNumber" : false,
        "byteLength" : 120,
        "calculated" : false,
        "calculatedFormula" : null,
        "cascadeDelete" : false,
        "caseSensitive" : false,
        "controllerName" : null,
        "createable" : true,
        "custom" : false,
        "defaultValue" : null,
        "defaultValueFormula" : null,
        "defaultedOnCreate" : false,
        "dependentPicklist" : false,
        "deprecatedAndHidden" : false,
        "digits" : 0,
        "displayLocationInDecimal" : false,
        "externalId" : false,
        "extraTypeInfo" : null,
```

```
        "filterable" : true,
        "groupable" : true,
        "htmlFormatted" : false,
        "idLookup" : false,
        "inlineHelpText" : null,
        "label" : "Account Phone",
        "length" : 40,
        "mask" : null,
        "maskType" : null,
        "name" : "Phone",
        "nameField" : false,
        "namePointing" : false,
        "nillable" : true,
        "permissionable" : true,
        "picklistValues" : [ ],
        "precision" : 0,
        "queryByDistance" : false,
        "referenceTo" : [ ],
        "relationshipName" : null,
        "relationshipOrder" : null,
        "restrictedDelete" : false,
        "restrictedPicklist" : false,
        "scale" : 0,
        "soapType" : "xsd:string",
        "sortable" : true,
        "type" : "phone",
        "unique" : false,
        "updateable" : true,
        "writeRequiresMasterRead" : false
      },
      "displayLines" : 1,
      "tabOrder" : 3,
      "type" : "Field",
      "value" : "Phone"
    } ],
    "placeholder" : false,
    "required" : false
  } ],
  "id" : "0AHD000000000AbOAI",
  "imageItems" : [ {
    "editable" : false,
    "label" : "Photo URL",
    "layoutComponents" : [ {
      "components" : [ ],
      "details" : {
        "autoNumber" : false,
        "byteLength" : 765,
        "calculated" : false,
        "calculatedFormula" : null,
        "cascadeDelete" : false,
        "caseSensitive" : false,
        "controllerName" : null,
        "createable" : false,
        "custom" : false,
```

```
        "defaultValue" : null,
        "defaultValueFormula" : null,
        "defaultedOnCreate" : false,
        "dependentPicklist" : false,
        "deprecatedAndHidden" : false,
        "digits" : 0,
        "displayLocationInDecimal" : false,
        "externalId" : false,
        "extraTypeInfo" : "imageurl",
        "filterable" : true,
        "groupable" : true,
        "htmlFormatted" : false,
        "idLookup" : false,
        "inlineHelpText" : null,
        "label" : "Photo URL",
        "length" : 255,
        "mask" : null,
        "maskType" : null,
        "name" : "PhotoUrl",
        "nameField" : false,
        "namePointing" : false,
        "nillable" : true,
        "permissionable" : false,
        "picklistValues" : [ ],
        "precision" : 0,
        "queryByDistance" : false,
        "referenceTo" : [ ],
        "relationshipName" : null,
        "relationshipOrder" : null,
        "restrictedDelete" : false,
        "restrictedPicklist" : false,
        "scale" : 0,
        "soapType" :
        "xsd:string",
        "sortable" : true,
        "type" : "url",
        "unique" : false,
        "updateable" : false,
        "writeRequiresMasterRead" : false
      },
      "displayLines" : 1,
      "tabOrder" : 1,
      "type" : "Field",
      "value" : "PhotoUrl"
    } ],
    "placeholder" : false,
    "required" : false
  } ],
  "label" : "Custom Account Compact Layout",
  "name" : "Custom_Account_Compact_Layout"
} ],
"defaultCompactLayoutId" : "0AHD000000000AbOAI",
"recordTypeCompactLayoutMappings" : [ {
  "available" : true,
```

```
    "compactLayoutId" : "0AHD000000000AbOAI",
    "compactLayoutName" : "Custom_Account_Compact_Layout",
    "recordTypeId" : "012000000000000AAA",
    "recordTypeName" : "Master",
    "urls" : {
      "compactLayout" :
"/services/data/v60.0/sobjects/Account/describe/compactLayouts/012000000000000AAA"
    }
  } ],
  "urls" : {
   "primary" : "/services/data/v60.0/sobjects/Account/describe/compactLayouts/primary"

  }
}
```


If you haven't defined any compact layouts for an object, the `compactLayoutId` returns as `Null`.

Return Headers Using sObject CompactLayouts

Returns only the headers that are returned by a GET request to the sObject CompactLayouts resource. This gives you a chance to see header values ahead of time before retrieving the content of the resource. This resource is available in REST API version 29.0 and later.

Syntax

URI

For a compact layout description for a specific object, use
`/services/data/vXX.X/sobjects/sObject/describe/compactLayouts/`

Formats

JSON, XML

HTTP methods

HEAD

Authentication

`Authorization: Bearer token`

Request parameters

None required

Example

Example Request

```
curl -X HEAD --head
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/describe/compactLayouts
 -H "Authorization: Bearer token"
```

sObject Layouts

Retrieves lists of page layouts and their descriptions. You can request information for all of a specific object's layouts or for layouts associated with a specified record type on a specific object.

IN THIS SECTION:

SEE ALSO:

Object Reference for the Salesforce Platform

Get Layouts and Descriptions for a Specified Object

Retrieves lists of layouts and their descriptions for a single object.

Syntax

URI

`/services/data/vXX.X/sobjects/sObject/describe/layouts/`

Formats

JSON, XML

HTTP Method

GET

Authentication

`Authorization: Bearer token`

Parameters

None required

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Battle_Station__c/describe/layouts/
 -H "Authorization: Bearer token"
```

Example Response Body

```
{
  "layouts" : [ {
    "buttonLayoutSection" : {
      "detailButtons" : [
          ...
      ]
    },
    "detailLayoutSections" : [
        ...
    ],
    "editLayoutSections" : [
```

```
        ...
    ],
    "feedView" : null,
    "highlightsPanelLayoutSection" : null,
    "id" : "00ho000000BKMebAAH",
    "multirowEditLayoutSections" : [ ],
    "offlineLinks" : [ ],
    "quickActionList" : {
      "quickActionListItems" : [
          ...
      ]
    },
    "relatedContent" : null,
    "relatedLists" : [
        ...
    ],
    "saveOptions" : [ ]
  } ],
  "recordTypeMappings" : [
      ...
  ],
  "recordTypeSelectorRequired" : [ false ]
}
```

Return Layout Headers for a Specified Object

Returns only the headers that are returned by a GET request to sObject Layouts resources. This gives you a chance to see header values ahead of time before retrieving the content of the resource.

Syntax

URI

/services/data/v***XX.X***/sobjects/***sObject***/describe/layouts/

Formats

JSON, XML

HTTP Method

HEAD

Authentication

Authorization: Bearer ***token***

Parameters

None required

Example

Example Request

```
curl -X HEAD --head
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Battle_Station__c/describe/layouts/
 -H "Authorization: Bearer token"
```

sObject Layouts for an Object With Multiple Record Types

Retrieves lists of page layouts and their descriptions for objects that have more than one record type defined.

Get Layouts and Descriptions for an Object With Multiple Record Types

Retrieves lists of layouts and their descriptions.

For a layout description for objects that have more than one record type defined, the *recordTypeId* can be obtained from the result from /services/data/v***XX.X***/sobjects/***sObject***/describe/layouts/

A GET request for objects that have more than one record type defined returns null for the layouts section of the response.

Syntax

URI

/services/data/v***XX.X***/sobjects/***sObject***/describe/layouts/***recordTypeId***

Formats

JSON, XML

HTTP Method

GET

Authentication

Authorization: Bearer ***token***

Parameters

None required

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Chocolate__c/describe/layouts/0125c000000oIN9AAM
 -H "Authorization: Bearer token"
```

Example Response Body

```
{
  "buttonLayoutSection" : {
    "detailButtons" : [
        ...
    ]
  },
  "detailLayoutSections" : [
        ...
  ],
  "editLayoutSections" : [
        ...
  ],
  "feedView" : null,
  "highlightsPanelLayoutSection" : null,
```

```
  "id" : "00ho000000CUJWIAA5",
  "multirowEditLayoutSections" : [ ],
  "offlineLinks" : [ ],
  "quickActionList" : {
    "quickActionListItems" : [
            ...
    ]
  },
  "relatedContent" : null,
  "relatedLists" : [
        ...
  ],
  "saveOptions" : [ ]
}
```

Return Layout Headers for an Object With Multiple Record Types

Returns only the headers that are returned by a GET request to sObject Layouts resources. This gives you a chance to see header values ahead of time before retrieving the content of the resource.

Syntax

URI

`/services/data/vXX.X/sobjects/sObject/describe/layouts/recordTypeId`

Formats

JSON, XML

HTTP Method

HEAD

Authentication

`Authorization: Bearer token`

Parameters

None required

Example

Example Request

```
curl -X HEAD
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Chocolate__c/describe/layouts/0125c000000oIN9AAM
 -H "Authorization: Bearer token"
```

sObject Global Publisher Layouts

Retrieves lists of global publisher layouts. Global publisher layouts customize the actions on global pages (like the Home page). In Lightning Experience, these layouts populate the Global Actions menu.

IN THIS SECTION:

Get Global Publisher Layouts and Descriptions

Retrieves lists of global publisher layouts and their descriptions. Global publisher layouts customize the actions on global pages (like the Home page). In Lightning Experience, these layouts populate the Global Actions menu.

Syntax

URI

`/services/data/vXX.X/sobjects/Global/describe/layouts/`

Formats

JSON, XML

HTTP Method

GET

Authentication

`Authorization: Bearer token`

Parameters

None required

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Global/describe/layouts/
 -H "Authorization: Bearer token"
```

Example Response Body

```
{
  "layouts": [
    {
      "buttonLayoutSection": null,
      "detailLayoutSections": [],
      "editLayoutSections": [],
      "feedView": null,
      "highlightsPanelLayoutSection": null,
      "id": "00hRO000000Mo6jYAC",
      "multirowEditLayoutSections": [],
      "offlineLinks": [],
      "quickActionList": {
        "quickActionListItems": [
```

```
          {
            "accessLevelRequired": null,
            "colors": [
              {
                "color": "65CAE4",
                "context": "primary",
                "theme": "theme4"
              }
            ],
            "iconUrl": null,
            "icons": [
              {
                "contentType": "image/svg+xml",
                "height": 0,
                "theme": "theme4",
                "url":
"https://rockyroads.test1.my.pc-rnd.salesforce.com/img/icon/t4v35/action/share_post.svg",

                "width": 0
              },
               ...
            ],
            "label": "Post",
            "miniIconUrl": "",
            "quickActionName": "FeedItem.TextPost",
            "targetSobjectType": null,
            "type": "Post",
            "urls": {}
          },
          ...
        ]
      },
      "relatedContent": null,
      "relatedLists": [],
      "saveOptions": []
    }
  ],
  "recordTypeMappings": [],
  "recordTypeSelectorRequired": [
    false
  ]
}
```

Return Headers for All Global Publisher Layouts

Returns only the headers that are returned by a GET request to sObject Layouts resources. This gives you a chance to see header values ahead of time before retrieving the content of the resource.

Syntax

URI

/services/data/v***XX.X***/sobjects/Global/describe/layouts/

Formats

JSON, XML

HTTP Method

HEAD

Authentication

`Authorization: Bearer token`

Parameters

None required

Example

Example Request

```
curl -X HEAD
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Global/describe/layouts/
 -H "Authorization: Bearer token"
```

sObject PlatformAction

Returns the description of the 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. This resource is available in API version 33.0 and later.

The only thing you can do with this resource is Query it.

Syntax

URI

`/services/data/vXX.X/sobjects/PlatformAction`

Formats

JSON, XML

HTTP methods

GET

Authentication

`Authorization: Bearer token`

Request body

None

SEE ALSO:

Object Reference for the Salesforce Platform

sObject Quick Actions

Access an object's actions and the action details. This resource is available in REST API version 28.0 and later.

When working with actions, also refer to Quick Actions.

IN THIS SECTION:

Get sObject Quick Actions

Returns a specific object's actions as well as global actions. This resource is available in REST API version 28.0 and later.

Return Headers Using sObject Quick Actions

Returns only the headers that are returned by sending a GET request to the sObject Quick Actions resource. This gives you a chance to see the header values before retrieving the content of the resource. This resource is available in REST API version 28.0 and later.

SEE ALSO:

Object Reference for the Salesforce Platform

Get sObject Quick Actions

Returns a specific object's actions as well as global actions. This resource is available in REST API version 28.0 and later.

This resource returns all actions—both global and standard—in addition to the ones requested.

When working with actions, also refer to Quick Actions.

Syntax

URI

/services/data/v***XX.X***/sobjects/***sObject***/quickActions/

Formats

JSON, XML

HTTP Method

GET

Authentication

Authorization: Bearer ***token***

Parameters

None required

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/quickActions/
 -H "Authorization: Bearer token"
```

Return Headers Using sObject Quick Actions

Returns only the headers that are returned by sending a GET request to the sObject Quick Actions resource. This gives you a chance to see the header values before retrieving the content of the resource. This resource is available in REST API version 28.0 and later.

This resource returns all actions—both global and standard—in addition to the ones requested.

When working with actions, also refer to Quick Actions.

Syntax

URI

/services/data/v***XX.X***/sobjects/***sObject***/quickActions/

Formats

JSON, XML

HTTP Method

HEAD

Authentication

Authorization: Bearer ***token***

Parameters

None required

Example

Example Request

```
curl -X HEAD --head
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/quickActions/
 -H "Authorization: Bearer token"
```

sObject Specific Quick Actions

Access a specific action for an object. By using the POST method with this resource, you can create records using an object's quick actions.

When working with actions, also refer to Quick Actions.

IN THIS SECTION:

Get Specific sObject Quick Actions

Gets a specific action for an object, as well as the action's details. This resource is available in REST API version 28.0 and later.

When working with actions, also refer to Quick Actions.

Syntax

URI

/services/data/v***XX.X***/sobjects/***sObject***/quickActions/***actionName***

Formats

JSON, XML

HTTP Method

GET

Authentication

Authorization: Bearer ***token***

Parameters

None required

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/quickActions/CreateContact
 -H "Authorization: Bearer token"
```

Create Records Using Specific sObject Quick Actions

Creates a record via the specified quick action based on the field values included in the request body. This resource is available in REST API version 28.0 and later.

When working with actions, also refer to Quick Actions.

Syntax

URI

/services/data/v***XX.X***/sobjects/***sObject***/quickActions/***actionName***

Formats

JSON, XML

HTTP Method

POST

Authentication

Authorization: Bearer ***token***

Parameters

None required

Example

Example Request

```
curl -X POST
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/quickActions/CreateContact
 -H 'Authorization: Bearer token -H "Content-Type: application/json" -d @newcontact.json'
```

Example Request Body

```
{

   "contextId" : "001D000000JRSGf",
   "record" : { "LastName" : "Smith" }

}
```

Return Headers Using Specific sObject Quick Actions

Returns only the headers that are returned by sending a GET request to the sObject Quick Actions resource. This gives you a chance to see the header values before retrieving the content of the resource. This resource is available in REST API version 28.0 and later.

When working with actions, also refer to Quick Actions.

Syntax

URI

`/services/data/vXX.X/sobjects/sObject/quickActions/actionName`

Formats

JSON, XML

HTTP Method

HEAD

Authentication

`Authorization: Bearer token`

Parameters

None required

Example

Example Request

```
curl -X HEAD --head
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/quickActions/CreateContact
 -H "Authorization: Bearer token"
```

sObject Quick Action Details

Access the descriptive detail for a specific action on an object.

When working with actions, also refer to Quick Actions.

IN THIS SECTION:

Get sObject Quick Action Details

Returns a specific action's descriptive detail. This resource is available in REST API version 28.0 and later.

Return Headers Using sObject Quick Action Details

Returns only the headers that are returned by sending a GET request to the sObject Quick Actions resource. This gives you a chance to see the header values before retrieving the content of the resource. This resource is available in REST API version 28.0 and later.

Get sObject Quick Action Details

Returns a specific action's descriptive detail. This resource is available in REST API version 28.0 and later.

When working with actions, also refer to Quick Actions.

Syntax

URI

`/services/data/vXX.X/sobjects/sObject/quickActions/actionName/describe/`

Formats

JSON, XML

HTTP Method

GET

Authentication

`Authorization: Bearer token`

Parameters

None required

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/quickActions/CreateContact/describe/
 -H "Authorization: Bearer token"
```

Return Headers Using sObject Quick Action Details

Returns only the headers that are returned by sending a GET request to the sObject Quick Actions resource. This gives you a chance to see the header values before retrieving the content of the resource. This resource is available in REST API version 28.0 and later.

When working with actions, also refer to Quick Actions.

Syntax

URI

`/services/data/vXX.X/sobjects/sObject/quickActions/actionName/describe/`

Formats

JSON, XML

HTTP Method
HEAD

Authentication
`Authorization: Bearer token`

Parameters
None required

Example

Example Request

```
curl -X HEAD --head
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/quickActions/CreateContact/describe/
 -H "Authorization: Bearer token"
```

sObject Quick Action Default Values

Retrieves the default values, including default field values, for a specific action on an object.

When working with actions, also refer to Quick Actions.

IN THIS SECTION:

Get sObject Quick Action Default Values
Returns a specific action's default values, including field values. This resource is available in REST API version 28.0 and later.

Return Headers Using sObject Quick Action Default Values
Returns only the headers that are returned by sending a GET request to the sObject Quick Actions resource. This gives you a chance to see the header values before retrieving the content of the resource. This resource is available in REST API version 28.0 and later.

Get sObject Quick Action Default Values

Returns a specific action's default values, including field values. This resource is available in REST API version 28.0 and later.

When working with actions, also refer to Quick Actions.

Syntax

URI
`/services/data/vXX.X/sobjects/sObject/quickActions/actionName/defaultValues/`

Formats
JSON, XML

HTTP Method
GET

Authentication
`Authorization: Bearer token`

Parameters
None required

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/quickActions/CreateContact/defaultValues/
 -H "Authorization: Bearer token"
```

Return Headers Using sObject Quick Action Default Values

Returns only the headers that are returned by sending a GET request to the sObject Quick Actions resource. This gives you a chance to see the header values before retrieving the content of the resource. This resource is available in REST API version 28.0 and later.

When working with actions, also refer to Quick Actions.

Syntax

URI

`/services/data/vXX.X/sobjects/sObject/quickActions/actionName/defaultValues/`

Formats

JSON, XML

HTTP Method

HEAD

Authentication

`Authorization: Bearer token`

Parameters

None required

Example

Example Request

```
curl -X HEAD --head
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/quickActions/CreateContact/defaultValues/
 -H "Authorization: Bearer token"
```

sObject Quick Action Default Values by ID

Evaluates the default values for a specific action on an object. Responses include default field values.

When working with actions, also refer to Quick Actions.

IN THIS SECTION:

[Return Headers Using sObject Quick Action Default Values by ID](#)
Returns only the headers that are returned by sending a GET request to the sObject Quick Actions resource. This gives you a chance to see the header values before retrieving the content of the resource. This resource is available in REST API version 29.0 and later.

Get sObject Quick Action Default Values by ID

Returns the default values for an action specific to the *`context_id`* object. This resource is available in REST API version 29.0 and later.

In API version 28.0, to evaluate the default values for an action, use `/services/data/vXX.X/sobjects/sObject/quickActions/action_name/defaultValues/parent_id`.

When working with actions, also refer to Quick Actions.

Syntax

URI

`/services/data/vXX.X/sobjects/sObject/quickActions/actionName/defaultValues/contextId`

Formats

JSON, XML

HTTP Method

GET

Authentication

`Authorization: Bearer token`

Parameters

None required

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/quickActions/CreateContact/defaultValues/001D000000JRWBd
 -H "Authorization: Bearer token"
```

Return Headers Using sObject Quick Action Default Values by ID

Returns only the headers that are returned by sending a GET request to the sObject Quick Actions resource. This gives you a chance to see the header values before retrieving the content of the resource. This resource is available in REST API version 29.0 and later.

In API version 28.0, to evaluate the default values for an action, use `/services/data/vXX.X/sobjects/sObject/quickActions/action_name/defaultValues/parent_id`.

When working with actions, also refer to Quick Actions.

Syntax

URI

`/services/data/vXX.X/sobjects/sObject/quickActions/actionName/defaultValues/contextId`

Formats

JSON, XML

HTTP Method

HEAD

Authentication

`Authorization: Bearer token`

Parameters

None required

Example

Example Request

```
curl -X HEAD --head
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/quickActions/CreateContact/defaultValues/001D000000JRWBd
 -H "Authorization: Bearer token"
```

sObject Rich Text Image Get

Gets the specified image data from a specific rich text area field in a given record. To get an image, you must have a record with an image uploaded to a rich text area field.

Syntax

URI

`/services/data/vXX.X/sobjects/sObject/id/richTextImageFields/fieldName/contentReferenceId`

Formats

Binary data

HTTP Method

GET

Authentication

`Authorization: Bearer token`

Parameters

Parameter	Description
sObjectName	Indicates the name of the standard object of the record.
id	The ID of the object.
fieldName	The name of the rich text area field.

Parameter	Description
`contentReferenceId`	The reference ID that uniquely identifies an image within a rich text area field. You can obtain the reference by retrieving information for the object. The description shows the contents of the rich text area field. For example:

```
{
   "attributes" : {
      "type" : "Lead",
      "url" :
"/services/data/v60.0/sobjects/Lead/00QRM000003ZfDb2AK"
   },
   "Id" : "00QRM000003ZfDb2AK",
   ...
   "ContactPhoto__c" :
   "Sarah Loehr and her two dogs.
   <img alt=\"Sarah Loehr.\"

src=\"https://MyDomainName.file.force.com/servlet/rtaImage?

   eid=00QRM000003ZfDb&amp;
   feoid=00NRM000001E73j&amp;
   refid=0EMRM00000002Ip\"></img>"
}
```

The `refid` parameter of the image (`0EMRM00000002Ip` in this example) is the `contentReferenceId`.

Example

For an example of retrieving the blob data from a rich text area field, see Get an Image from a Rich Text Area Field on page 72.

SEE ALSO:

Object Reference for the Salesforce Platform

sObject Relationships

Accesses records by traversing object relationships via friendly URLs. You can retrieve, update, or delete the record associated with the traversed relationship field. If there are multiple related records, you can retrieve the complete set of associated records. This resource is available in REST API version 36.0 and later.

IN THIS SECTION:

Get Records Using sObject Relationships

Gets a record based on the specified object, record ID, and relationship field. The fields and field values of the record are returned in the response body. If there are multiple related records, you can retrieve the complete set of associated records.

Update Records Using sObject Relationships

Updates a parent record based on the specified object, record ID, and relationship field name. Field values provided in the request body replace the existing values in the record. Only a child-to-parent relationship can be traversed when you update records.

Delete Records Using sObject Relationships

Deletes a parent record based on the specified object, record ID, and relationship field name. Only a child-to-parent relationship can be traversed when you delete records.

Get Records Using sObject Relationships

Gets a record based on the specified object, record ID, and relationship field. The fields and field values of the record are returned in the response body. If there are multiple related records, you can retrieve the complete set of associated records.

If there's no record associated with a relationship field, a 404 error response is returned. If the relationship field normally resolves to multiple records and no relationship set exists, a 200 response is returned. If the `fields` parameter is used with fields that don't exist or aren't visible to the consumer by field-level security, a 400 error response is returned. For other error messages, see Status Codes and Error Responses.

Syntax

URI

`/services/data/vXX.X/sobjects/sObject/id/relationshipFieldName`

Formats

JSON, XML

HTTP Method

GET

Authentication

`Authorization: Bearer token`

Parameters

Parameter	Description
`sObject`	The name of the object. For example, `Account`.
`id`	The identifier of the record. For example, `001R0000005hDFYIA2`.
`relationshipFieldFame`	The name of the field that contains the relationship. For example, `Opportunities`.
`fields`	Optional for GET. A comma-delimited list of fields in the associated relationship record returned in the response body. For example: `/services/data/v60.0/sobjects/sObject/id/relationship field?fields=field1,field2`

Example

For examples of using sObject Relationships to access relationship fields, see Traverse Relationships with Friendly URLs.

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Merchandise__c/a01D000000INjVe/Distributor__r
 -H "Authorization: Bearer token"
```

Example Response Body

The response body is the contents of the record associated with the relationship field. Here's an example of a request and JSON response body for a simple relationship traversal that returns the Distributor__c record associated with a relationship field on custom object Merchandise__c.

```
{
    "attributes" :
    {
        "type" : "Distributor__c",
        "url" : "/services/data/v60.0/sobjects/Distributor__c/a03D0000003DUhcIAG"
    },
    "Id" : "a03D0000003DUhcIAG",
    "OwnerId" : "005D0000001KyEIIA0",
    "IsDeleted" : false,
    "Name" : "Distributor1",
    "CreatedDate" : "2011-12-16T17:43:01.000+0000",
    "CreatedById" : "005D0000001KyEIIA0",
    "LastModifiedDate" : "2011-12-16T17:43:01.000+0000",
    "LastModifiedById" : "005D0000001KyEIIA0",
    "SystemModstamp" : "2011-12-16T17:43:01.000+0000",
    "Location__c" : "San Francisco"
}
```

SEE ALSO:

Object Reference for the Salesforce Platform

Update Records Using sObject Relationships

Updates a parent record based on the specified object, record ID, and relationship field name. Field values provided in the request body replace the existing values in the record. Only a child-to-parent relationship can be traversed when you update records.

If the `fields` parameter is used with fields that don't exist or aren't visible to the consumer by field-level security, a 400 error response is returned. For other error messages, see Status Codes and Error Responses.

Syntax

URI

`/services/data/vXX.X/sobjects/sObject/id/relationshipFieldName`

Formats

JSON, XML

HTTP Method

PATCH

Authentication

`Authorization: Bearer token`

Parameters

Parameter	Description
`sObject`	The name of the object. For example, `Contact`.
`id`	The identifier of the record. For example, `003R0000005hDFYIA2`, the contact ID.
`relationshipFieldName`	The name of the field that contains the relationship. For example, `Account`. Account is the name of the relationship on the child Contact object.

Example

For an example of updating a record using PATCH, see

Update a Record.

SEE ALSO:

Object Reference for the Salesforce Platform

Delete Records Using sObject Relationships

Deletes a parent record based on the specified object, record ID, and relationship field name. Only a child-to-parent relationship can be traversed when you delete records.

If the `fields` parameter is used with fields that don't exist or aren't visible to the consumer by field-level security, a 400 error response is returned. For other error messages, see Status Codes and Error Responses.

Syntax

URI

`/services/data/vXX.X/sobjects/sObject/id/relationshipFieldName`

Formats

JSON, XML

HTTP Method

DELETE

Authentication

`Authorization: Bearer token`

Parameters

Parameter	Description
`sObject`	The name of the object. For example, `Contact`.
`id`	The identifier of the record. For example, `003R0000005hDFYIA2`, the contact ID.

Parameter	Description
`relationshipFieldName`	The name of the field that contains the relationship. For example, `Account`. Account is the name of the relationship on the child Contact object.

When you delete a parent record, it deletes all child records that have a master-detail relationship to the parent record.

Example

For examples of using sObject Relationships to delete a relationship record, see Traverse Relationships with Friendly URLs.

SEE ALSO:

Object Reference for the Salesforce Platform

sObjects Suggested Articles

Returns results on suggested articles for a Case, Work Order, or Work Order Line. These suggestions are based on common keywords in the title, description, and other information that's entered before the record has been saved and assigned an ID. This resource is available in REST API version 30.0 and later.

Salesforce Knowledge must be enabled in your organization. The user must have the "View Articles" permission enabled. The articles suggested include only the articles the user can access, based on the data categories and article types the user has permissions to view.

Articles are suggested based on a relevance algorithm. The `suggestedArticles` resource is designed to get the IDs of articles relevant to a case, work order, or work order line item. It's intended to be used with other services that than use the IDs to get article data for display.

Syntax

URI

`/services/data/vXX.X/sobjects/sObject/suggestedArticles ?language=articleLanguage&subject=subject&description=description.`

Formats

JSON, XML

HTTP methods

GET

Authentication

`Authorization: Bearer token`

Request body

None required

Request parameters

Parameter	Description
`articleTypes`	Optional. Three-character ID prefixes indicating the desired article types. You can specify multiple values for this parameter in a single REST call, by repeating the parameter name for each value. For example, `articleTypes=ka0&articleTypes=ka1`.
`categories`	Optional. The name of the data category group and the data category API name (not category title) for desired articles. The syntax is `categories={"Group":"Category"}`. Characters in the URL might need to be encoded. For example: `categories=%7B%22Regions%22%3A%22Asia%22%2C%22 Products%22%3A%22Laptops%22%7D` The same data category group can't be specified more than once. However, you can specify multiple data category group and data category pairs. For example, `categories={"Regions":"Asia","Products":"Laptops"}`.
`description`	Text of the description. Valid only for new records without an existing ID and required if `subject` is null. Article suggestions are based on common keywords in the subject, description, or both.
`language`	Required. Language that the article is written in.
`limit`	Optional. Specifies the maximum number of suggested articles to return.
`publishStatus`	Optional. The article's publication status. Valid values: • `Draft`—Not published • `Online`—Published in Salesforce Knowledge • `Archived`
`subject`	Text of the subject. Valid only for new records without an existing ID and required if `description` is null. Article suggestions are based on common keywords in the subject, description, or both.
`topics`	Optional. The topic of returned articles. For example: `topics=outlook&topics=email`.
`validationStatus`	Optional. The validation status of returned articles.

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Case/suggestedArticles?
language=en_US&subject=orange+banana&articleTypes=ka0&articleTypes=ka1
 -H "Authorization: Bearer token"
```

Example Response Body

```
[ {
  "attributes" : {
    "type" : "KnowledgeArticleVersion",
    "url" : "/services/data/v60.0/sobjects/KnowledgeArticleVersion/ka0D00000004CcQ"
  "Id" : "ka0D00000004CcQ"
}, {
  "attributes" : {
    "type" : "KnowledgeArticleVersion",
    "url" : "/services/data/v60.0/sobjects/KnowledgeArticleVersion/ka0D00000004CXo"
  },
  "Id" : "ka0D00000004CXo"
} ]
```

sObjects Suggested Articles by ID

When you enter an article ID, you can retrieve records that offer similar information as the ID you entered. This resource is available in REST API version 30.0 and later.

Salesforce Knowledge must be enabled in your organization. The user must have the “View Articles” permission enabled. The articles suggested include only the articles the user can access, based on the data categories and article types the user has permissions to view.

Articles are suggested based on a relevance algorithm. The `suggestedArticles` resource is designed to get the IDs of articles relevant to a case, work order, or work order line item. It’s intended to be used with other services that than use the IDs to get article data for display.

Syntax

URI

`/services/data/vXX.X/sobjects/sObject/ID/suggestedArticles?language=articleLanguage`

Formats

JSON, XML

HTTP methods

GET

Authentication

`Authorization: Bearer token`

Request body

None required

Request parameters

Parameter	Description
`articleTypes`	Optional. Three-character ID prefixes indicating the desired article types. You can specify multiple values for this parameter in a single REST call, by repeating the parameter name for each value. For example, `articleTypes=ka0&articleTypes=ka1`.
`categories`	Optional. The name of the data category group and the data category API name (not category title) for desired articles. The syntax is

Parameter	Description
	`categories={"Group":"Category"}`. Characters in the URL might need to be encoded. For example: `categories=%7B%22Regions%22%3A%22Asia%22%2C%22 Products%22%3A%22Laptops%22%7D` The same data category group can't be specified more than one time. However, you can specify multiple data category group and data category pairs. For example, `categories={"Regions":"Asia","Products":"Laptops"}`.
`description`	Text of the description. Valid only for new records without an existing ID and required if `subject` is null. Article suggestions are based on common keywords in the subject, description, or both.
`language`	Required. Language that the article is written in.
`limit`	Optional. Specifies the maximum number of suggested articles to return.
`publishStatus`	Optional. The article's publication status. Valid values: • `Draft`—Not published • `Online`—Published in Salesforce Knowledge • `Archived`
`subject`	Text of the subject. Valid only for new records without an existing ID and required if `description` is null. Article suggestions are based on common keywords in the subject, description, or both.
`topics`	Optional. The topic of returned articles. For example: `topics=outlook&topics=email`.
`validationStatus`	Optional. The validation status of returned articles.

Example

Example Response Body

```
[ {
  "attributes" : {
    "type" : "KnowledgeArticleVersion",
    "url" : "/services/data/v60.0/sobjects/KnowledgeArticleVersion/ka0D00000004CcQ"
  "Id" : "ka0D00000004CcQ"
}, {
  "attributes" : {
    "type" : "KnowledgeArticleVersion",
    "url" : "/services/data/v60.0/sobjects/KnowledgeArticleVersion/ka0D00000004CXo"
  },
  "Id" : "ka0D00000004CXo"
} ]
```

sObject User Password

Accesses user passwords based on the specified user ID. Sets, resets, or gets the expiration status of a user password based on the HTTP method. Use the GET method to retrieve a password's expiration status, the POST method to set a password, or the DELETE method to initiate a password reset.

IN THIS SECTION:

Get User Password Expiration Status

Gets a user's password expiration status based on the specified user ID. A True or False value is returned in the response body. This resource is available in REST API version 24.0 and later.

If the session doesn't have permission to access the user information, an INSUFFICIENT_ACCESS error is returned.

Syntax

URI

`/services/data/vXX.X/sobjects/User/userId/password`

Formats

JSON, XML

HTTP Method

GET

Authentication

`Authorization: Bearer token`

Parameters

None required

Example

For examples of getting password information, setting a password, and resetting a password, see Manage User Passwords on page 82.

SEE ALSO:

Object Reference for the Salesforce Platform

Set User Password

Sets a user's password based on the specified user ID. The password provided in the request body replaces the user's existing password. This resource is available in REST API version 24.0 and later.

You can only set one password per request. The new password must conform to the password policies for the organization, otherwise an INVALID_NEW_PASSWORD error is returned.

If the session doesn't have permission to access the user information, an INSUFFICIENT_ACCESS error is returned.

Syntax

URI

`/services/data/vXX.X/sobjects/User/userId/password`

Formats

JSON, XML

HTTP Method

POST

Authentication

`Authorization: Bearer token`

Parameters

None required

Example

For examples of getting password information, setting a password, and resetting a password, see Manage User Passwords on page 82.

SEE ALSO:

Object Reference for the Salesforce Platform

Reset User Password

Initiates a password reset for a user based on the specified user ID. The user's current password becomes invalid and the user receives an email with a password reset link. To log in again, the user must finish resetting their password. This resource is available in REST API version 24.0 and later.

Salesforce auto-generates a temporary password for the user and returns it in the response body. If the user can't access the email link, they can finish resetting their password by logging in with the temporary password.

If the session doesn't have permission to access the user information, an INSUFFICIENT_ACCESS error is returned.

Syntax

URI

`/services/data/vXX.X/sobjects/User/userId/password`

Formats

JSON, XML

HTTP Method

DELETE

Authentication

`Authorization: Bearer token`

Parameters

None required

Example

For examples of getting password information, setting a password, and resetting a password, see Manage User Passwords on page 82.

SEE ALSO:

Object Reference for the Salesforce Platform

Return Headers Using sObject User Password

Returns only the headers that are returned by sending a GET request to the sObject User Password resource. This operation allows you to see returned header values of the GET request before retrieving the content itself. This resource is available in REST API version 24.0 and later.

If the session doesn't have permission to access the user information, an INSUFFICIENT_ACCESS error is returned.

Syntax

URI

`/services/data/vXX.X/sobjects/User/userId/password`

Formats

JSON, XML

HTTP Method

HEAD

Authentication

`Authorization: Bearer token`

Parameters

None required

Example

For examples of getting password information, setting a password, and resetting a password, see Manage User Passwords on page 82.

SEE ALSO:

Object Reference for the Salesforce Platform

sObject Self-Service User Password

Accesses self-service user passwords based on the specified user ID. Sets, resets, or gets the expiration status of a self-service user password based on the HTTP method. Use the GET method to retrieve a password's expiration status, the POST method to set a password, or the DELETE method to initiate a password reset.

IN THIS SECTION:

Get Self-Service User Password Expiration Status

Retrieves a self-service user's password expiration status based on the specified user ID. A True or False value is returned in the response body. This resource is available in REST API version 24.0 and later.

Set Self-Service User Password

Sets a self-service user's password based on the specified user ID. The password provided in the request body replaces the user's existing password. This resource is available in REST API version 24.0 and later.

Reset Self-Service User Password

Initiates a password reset for a self-service user based on the specified user ID. The user's current password becomes invalid and the user receives an email with a password reset link. To log in again, the user must finish resetting their password. This resource is available in REST API version 24.0 and later.

Return Headers Using sObject Self-Service User Password

Returns only the headers that are returned by sending a GET request to the sObject Self-Service User Password resource. This operation allows you to see returned header values of the GET request before retrieving the content itself. This resource is available in REST API version 24.0 and later.

Get Self-Service User Password Expiration Status

Retrieves a self-service user's password expiration status based on the specified user ID. A True or False value is returned in the response body. This resource is available in REST API version 24.0 and later.

If the session doesn't have permission to access the user information, an INSUFFICIENT_ACCESS error is returned.

Syntax

URI

`/services/data/vXX.X/sobjects/SelfServiceUser/selfServiceUserId/password`

Formats

JSON, XML

HTTP Method

GET

Authentication

`Authorization: Bearer token`

Parameters

None required

Example

For examples of getting password information, setting a password, and resetting a password, see Manage User Passwords on page 82.

SEE ALSO:

Object Reference for the Salesforce Platform

Set Self-Service User Password

Sets a self-service user's password based on the specified user ID. The password provided in the request body replaces the user's existing password. This resource is available in REST API version 24.0 and later.

You can only set one password per request. The new password must conform to the password policies for the organization, otherwise an INVALID_NEW_PASSWORD error is returned.

If the session doesn't have permission to access the user information, an INSUFFICIENT_ACCESS error is returned.

Syntax

URI

`/services/data/vXX.X/sobjects/SelfServiceUser/selfServiceUserId/password`

Formats

JSON, XML

HTTP Method

POST

Authentication

`Authorization: Bearer token`

Parameters

None required

Example

For examples of getting password information, setting a password, and resetting a password, see Manage User Passwords on page 82.

SEE ALSO:

Object Reference for the Salesforce Platform

Reset Self-Service User Password

Initiates a password reset for a self-service user based on the specified user ID. The user's current password becomes invalid and the user receives an email with a password reset link. To log in again, the user must finish resetting their password. This resource is available in REST API version 24.0 and later.

Salesforce auto-generates a temporary password for the user and returns it in the response body. If the user can't access the email link, they can finish resetting their password by logging in with the temporary password.

If the session doesn't have permission to access the user information, an INSUFFICIENT_ACCESS error is returned.

Syntax

URI

`/services/data/vXX.X/sobjects/SelfServiceUser/selfServiceUserId/password`

Formats

JSON, XML

HTTP Method

DELETE

Authentication

`Authorization: Bearer token`

Parameters

None required

Example

For examples of getting password information, setting a password, and resetting a password, see Manage User Passwords on page 82.

SEE ALSO:

Object Reference for the Salesforce Platform

Return Headers Using sObject Self-Service User Password

Returns only the headers that are returned by sending a GET request to the sObject Self-Service User Password resource. This operation allows you to see returned header values of the GET request before retrieving the content itself. This resource is available in REST API version 24.0 and later.

If the session doesn't have permission to access the user information, an INSUFFICIENT_ACCESS error is returned.

Syntax

URI

`/services/data/vXX.X/sobjects/SelfServiceUser/selfServiceUserId/password`

Formats

JSON, XML

HTTP Method

HEAD

Authentication

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

Parameters

None required

Example

For examples of getting password information, setting a password, and resetting a password, see Manage User Passwords on page 82.

SEE ALSO:

Object Reference for the Salesforce Platform

Platform Event Schema by Event Name

Gets the definition of a platform event in JSON format for an event name. This resource is available in REST API version 40.0 and later.

400 Bad Request	Description
In API version 43.0 and later	The request was formatted incorrectly—an invalid value was passed for the `payloadFormat` parameter in the URI.
In API version 42.0 and earlier	The request was formatted incorrectly—the `payloadFormat` parameter was passed in the URI but this API version doesn't support this parameter.

Syntax

URI

`/services/data/v`***`XX.X`***`/sobjects/`***`eventName`***`/eventSchema`

Formats

JSON

HTTP methods

GET

Authentication

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

Parameters

Parameter	Description
`payloadFormat`	(Optional query parameter. Available in API version 43.0 and later.) The format of the returned event schema. This parameter can take one of the following values. • `EXPANDED`—The JSON representation of the event schema, which is the default format when `payloadFormat` isn't specified in API version 43.0 and later. The expanded event schema is in the open-source Apache Avro format but doesn't strictly adhere to the record complex type. For more information about the schema fields, see Apache Avro Format.

Parameter	Description
	• COMPACT—The JSON representation of the event schema that adheres to the open-source Apache Avro specification for the record complex type. For more information about the schema fields, see Apache Avro Format. Subscribers use the compact schema format to deserialize compact events received in binary form.

Examples for API Version 43.0 and Later

This URI gets the schema of a platform event named `Low_Ink__e`. In API version 43.0 and later, the default response format is the JSON representation of the event schema.

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Low_Ink__e/eventSchema
 -H "Authorization: Bearer token"
```

Or:

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Low_Ink__e/eventSchema?payloadFormat=EXPANDED
 -H "Authorization: Bearer token"
```

The returned response for the expanded format looks like the following in API version 60.0.

```
{
  "name": "Low_Ink__e",
  "namespace": "com.sforce.eventbus",
  "type": "expanded-record",
  "fields": [
    {
      "name": "data",
      "type": {
        "type": "record",
        "name": "Data",
        "namespace": "",
        "fields": [
          {
            "name": "schema",
            "type": "string"
          },
          {
            "name": "payload",
            "type": {
              "type": "record",
              "name": "Payload",
              "doc": "",
              "fields": [
                {
                  "name": "CreatedDate",
                  "type": "string",
                  "doc": "CreatedDate:DateTime"
                },
                {
```

```
              "name": "CreatedById",
              "type": "string",
              "doc": "CreatedBy:EntityId"
            },
            {
              "name": "Printer_Model__c",
              "type": [
                "null",
                "string"
              ],
              "doc": "Data:Text:00NRM000001krnv",
              "default": null
            },
            {
              "name": "Serial_Number__c",
              "type": [
                "null",
                "string"
              ],
              "doc": "Data:Text:00NRM000001kro0",
              "default": null
            },
            {
              "name": "Ink_Percentage__c",
              "type": [
                "null",
                "double"
              ],
              "doc": "Data:Double:00NRM000001kro5",
              "default": null
            }
          ]
        }
      },
      {
        "name": "event",
        "type": {
          "type": "record",
          "name": "Event",
          "fields": [
            {
              "name": "replayId",
              "type": "long"
            }
          ]
        }
      }
    ]
  }
},
{
  "name": "channel",
  "type": "string"
}
```

```
  ]
}
```

To get the compact (Apache Avro) format, use the following URI.

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Low_Ink__e/eventSchema?payloadFormat=COMPACT
 -H "Authorization: Bearer token"
```

The returned response for the compact format looks like the following in API version 60.0.


```
{
  "name": "Low_Ink__e",
  "namespace": "com.sforce.eventbus",
  "type": "record",
  "fields": [
    {
      "name": "CreatedDate",
      "type": "long",
      "doc": "CreatedDate:DateTime"
    },
    {
      "name": "CreatedById",
      "type": "string",
      "doc": "CreatedBy:EntityId"
    },
    {
      "name": "Printer_Model__c",
      "type": [
        "null",
        "string"
      ],
      "doc": "Data:Text:00NRM000001krnv",
      "default": null
    },
    {
      "name": "Serial_Number__c",
      "type": [
        "null",
        "string"
      ],
      "doc": "Data:Text:00NRM000001kro0",
      "default": null
    },
    {
      "name": "Ink_Percentage__c",
      "type": [
        "null",
        "double"
      ],
      "doc": "Data:Double:00NRM000001kro5",
      "default": null
    }
  ],
```

```
  "uuid": "5E5OtZj5_Gm6Vax9XMXH9A"
}
```

Note: The compact schema doesn't include the `replayId` or `channel` fields because these fields aren't necessary for deserializing the compact event received.

Examples for API Version 42.0 and Earlier

In API version 42.0 and earlier, the response format adheres to the open-source Apache Avro specification for the record complex type.

Note: This format is what the API returns in API version 43.0 and later when appending the `?payloadFormat=COMPACT` parameter.

```
curl
https://MyDomainName.my.salesforce.com/services/data/v42.0/sobjects/Low_Ink__e/eventSchema
 -H "Authorization: Bearer token"
```

The returned response looks like the following in API version 42.0.


```
{
  "name": "Low_Ink__e",
  "namespace": "com.sforce.eventbus",
  "type": "record",
  "fields": [
    {
      "name": "CreatedDate",
      "type": "long",
      "doc": "CreatedDate:DateTime"
    },
    {
      "name": "CreatedById",
      "type": "string",
      "doc": "CreatedBy:EntityId"
    },
    {
      "name": "Printer_Model__c",
      "type": [
        "null",
        "string"
      ],
      "doc": "Data:Text:00NRM000001krnv",
      "default": null
    },
    {
      "name": "Serial_Number__c",
      "type": [
        "null",
        "string"
      ],
      "doc": "Data:Text:00NRM000001kro0",
      "default": null
    },
    {
      "name": "Ink_Percentage__c",
```

```
      "type": [
        "null",
        "double"
      ],
      "doc": "Data:Double:00NRM000001kro5",
      "default": null
    }
  ],
  "uuid": "5E5OtZj5_Gm6Vax9XMXH9A"
}
```

Note: When you change the definition of a platform event, the schema ID for this platform event changes.

Apache Avro Format

The fields in the returned response adhere to the open-source Apache Avro specification for the record complex type (see Avro Records in the Apache Avro specification). Note the following:

- `name` is the name of the platform event.
- `namespace` corresponds to `com.sforce.eventbus`.
- `type` is the Avro complex type.
- `fields` is a JSON array containing the fields of the platform event. For each field:
  - `type` indicates that the field can be either null or have a value of the specified type. When the field isn't present, the value is `default`.
  - `doc` describes the field data type and includes the field ID for custom fields. This field is intended for internal use. For example, Salesforce uses the data type information to convert DateTime fields from long to DateTime. We recommend that you don't rely on this field's value because it might change in the future.

The response also includes the `uuid` field, which contains the schema's ID. The ID is the MD5 fingerprint of the normalized Avro schema encoded as a base-64 URL variant. You can append this ID to the `/vXX.X/event/eventSchema/` URI to retrieve the schema.

Platform Event Schema by Schema ID

Gets the definition of a platform event in JSON format for a schema ID. This resource is available in REST API version 40.0 and later.

400 Bad Request	Description
In API version 43.0 and later	The request was formatted incorrectly—an invalid value was passed for the `payloadFormat` parameter in the URI.
In API version 42.0 and earlier	The request was formatted incorrectly—the `payloadFormat` parameter was passed in the URI but this API version doesn't support this parameter.

Syntax

URI

`/services/data/vXX.X/event/eventSchema/schemaId`

Formats

JSON

HTTP methods

GET

Authentication

`Authorization: Bearer token`

Parameters

Parameter	Description
`payloadFormat`	(Optional query parameter. Available in API version 43.0 and later.) The format of the returned event schema. This parameter can take one of the following values. • `EXPANDED`—The JSON representation of the event schema, which is the default format when `payloadFormat` isn't specified in API version 43.0 and later. The expanded event schema is in the open-source Apache Avro format but doesn't strictly adhere to the record complex type. For more information about the schema fields, see Apache Avro Format. • `COMPACT`—The JSON representation of the event schema that adheres to the open-source Apache Avro specification for the record complex type. For more information about the schema fields, see Apache Avro Format. Subscribers use the compact schema format to deserialize compact events received in binary form.

Examples for API Version 43.0 and Later

This URI gets the schema of a platform event whose schema ID is `5E5OtZj5_Gm6Vax9XMXH9A`. This schema ID is a sample ID. Replace it with a valid schema ID for your event.

```
/services/data/v60.0/event/eventSchema/5E5OtZj5_Gm6Vax9XMXH9A
```

Or:

```
/services/data/v60.0/event/eventSchema/5E5OtZj5_Gm6Vax9XMXH9A?payloadFormat=EXPANDED
```

In API version 43.0 and later, the response format is the JSON representation of the event schema by default. The returned response looks like the following in API version 60.0.

```
{
  "name": "Low_Ink__e",
  "namespace": "com.sforce.eventbus",
  "type": "expanded-record",
  "fields": [
    {
      "name": "data",
      "type": {
        "type": "record",
        "name": "Data",
        "namespace": "",
        "fields": [
          {
            "name": "schema",
            "type": "string"
```

```
        },
        {
          "name": "payload",
          "type": {
            "type": "record",
            "name": "Payload",
            "doc": "",
            "fields": [
              {
                "name": "CreatedDate",
                "type": "string",
                "doc": "CreatedDate:DateTime"
              },
              {
                "name": "CreatedById",
                "type": "string",
                "doc": "CreatedBy:EntityId"
              },
              {
                "name": "Printer_Model__c",
                "type": [
                  "null",
                  "string"
                ],
                "doc": "Data:Text:00NRM000001krnv",
                "default": null
              },
              {
                "name": "Serial_Number__c",
                "type": [
                  "null",
                  "string"
                ],
                "doc": "Data:Text:00NRM000001kro0",
                "default": null
              },
              {
                "name": "Ink_Percentage__c",
                "type": [
                  "null",
                  "double"
                ],
                "doc": "Data:Double:00NRM000001kro5",
                "default": null
              }
            ]
          }
        },
        {
          "name": "event",
          "type": {
            "type": "record",
            "name": "Event",
            "fields": [
```

```
                  {
                    "name": "replayId",
                    "type": "long"
                  }
                ]
              }
            }
          ]
        }
      },
      {
        "name": "channel",
        "type": "string"
      }
    ]
}
```

To get the compact (Apache Avro) format, use the following URI.

```
/services/data/v60.0/event/eventSchema/5E5OtZj5_Gm6Vax9XMXH9A?payloadFormat=COMPACT
```

The returned response for the compact format looks like the following in API version 60.0.

```
{
  "name": "Low_Ink__e",
  "namespace": "com.sforce.eventbus",
  "type": "record",
  "fields": [
    {
      "name": "CreatedDate",
      "type": "long",
      "doc": "CreatedDate:DateTime"
    },
    {
      "name": "CreatedById",
      "type": "string",
      "doc": "CreatedBy:EntityId"
    },
    {
      "name": "Printer_Model__c",
      "type": [
        "null",
        "string"
      ],
      "doc": "Data:Text:00NRM000001krnv",
      "default": null
    },
    {
      "name": "Serial_Number__c",
      "type": [
        "null",
        "string"
      ],
      "doc": "Data:Text:00NRM000001kro0",
      "default": null
```

```
    },
    {
      "name": "Ink_Percentage__c",
      "type": [
        "null",
        "double"
      ],
      "doc": "Data:Double:00NRM000001kro5",
      "default": null
    }
  ],
  "uuid": "5E5OtZj5_Gm6Vax9XMXH9A"
}
```

Note: The compact schema doesn't include the `replayId` or `channel` fields because these fields aren't necessary for deserializing the compact event received.

Example for API Version 42.0 and Earlier

In API version 42.0 and earlier, the response format adheres to the open-source Apache Avro specification for the record complex type.

Note: This format is what the API returns in API version 43.0 and later when appending the `?payloadFormat=COMPACT` parameter.

This URI gets the schema of a platform event whose schema ID is `5E5OtZj5_Gm6Vax9XMXH9A`. This schema ID is a sample ID. Replace it with a valid schema ID for your event.

```
/services/data/v42.0/event/eventSchema/5E5OtZj5_Gm6Vax9XMXH9A
```

The returned response looks like the following in API version 42.0.

```
{
  "name": "Low_Ink__e",
  "namespace": "com.sforce.eventbus",
  "type": "record",
  "fields": [
    {
      "name": "CreatedDate",
      "type": "long",
      "doc": "CreatedDate:DateTime"
    },
    {
      "name": "CreatedById",
      "type": "string",
      "doc": "CreatedBy:EntityId"
    },
    {
      "name": "Printer_Model__c",
      "type": [
        "null",
        "string"
      ],
      "doc": "Data:Text:00NRM000001krnv",
      "default": null
    },
```

```
    {
      "name": "Serial_Number__c",
      "type": [
        "null",
        "string"
      ],
      "doc": "Data:Text:00NRM000001kro0",
      "default": null
    },
    {
      "name": "Ink_Percentage__c",
      "type": [
        "null",
        "double"
      ],
      "doc": "Data:Double:00NRM000001kro5",
      "default": null
    }
  ],
  "uuid": "5E5OtZj5_Gm6Vax9XMXH9A"
}
```


Note: When you change the definition of a platform event, the schema ID for this platform event changes.

If you don't have the schema ID, you can get the schema by supplying the platform event name. Make a GET request to `/services/data/vXX.X/sobjects/eventName/eventSchema`. See Platform Event Schema by Event Name.

Apache Avro Format

The fields in the returned response adhere to the open-source Apache Avro specification for the record complex type (see Avro Records in the Apache Avro specification). Note the following:

- `name` is the name of the platform event.
- `namespace` corresponds to `com.sforce.eventbus`.
- `type` is the Avro complex type.
- `fields` is a JSON array containing the fields of the platform event. For each field:
  - `type` indicates that the field can be either null or have a value of the specified type. When the field isn't present, the value is `default`.
  - `doc` describes the field data type and includes the field ID for custom fields. This field is intended for internal use. For example, Salesforce uses the data type information to convert DateTime fields from long to DateTime. We recommend that you don't rely on this field's value because it might change in the future.

The response also includes the `uuid` field, which contains the schema's ID. The ID is the MD5 fingerprint of the normalized Avro schema encoded as a base-64 URL variant. You can append this ID to the `/vXX.X/event/eventSchema/` URI to retrieve the schema.

Get AppMenu Types

Gets a list of App Menu types in the Salesforce app dropdown menu. This resource is available in REST API version 29.0 and later.

Syntax

URI

`/services/data/vXX.X/appMenu/`

Formats

JSON, XML

HTTP method

GET

Authentication

`Authorization: Bearer token`

Request body

None

Request parameters

None required

Example

Example Request

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/appMenu/ -H
"Authorization: Bearer token"
```

AppMenu Items

Accesses App Menu items from the Salesforce app dropdown menu. To retrieve a list of App Menu items, use the GET method. To retrieve the headers returned by a request for App Menu items, use the HEAD method.

IN THIS SECTION:

Get AppMenu Items

Gets a list of the App Menu items in the Salesforce Lightning dropdown menu. This resource is available in REST API version 29.0 and later.

Syntax

URI

`/services/data/vXX.X/appMenu/AppSwitcher/`

Formats
JSON, XML

HTTP methods
GET

Authentication
`Authorization: Bearer token`

Request body
None

Request parameters
None required

Example

Example Request

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/appMenu/AppSwitcher -H
"Authorization: Bearer token"
```

Return Headers of App Menu Item Requests

Returns only the headers that are returned by a GET request for the Salesforce app dropdown menu items. Use this URI to see the header values before you retrieve the content of the resource. This resource is available in REST API version 29.0 and later.

Syntax

URI
`/services/data/vXX.X/appMenu/AppSwitcher/`

Formats
JSON, XML

HTTP method
HEAD

Authentication
`Authorization: Bearer token`

Request body
None

Request parameters
None required

Example

Example Request

```
curl -X HEAD --head
https://MyDomainName.my.salesforce.com/services/data/v60.0/appMenu/AppSwitcher -H
"Authorization: Bearer token"
```

AppMenu Mobile Items

Accesses App Menu items from the Salesforce mobile app for Android and iOS and the mobile web navigation menu.

IN THIS SECTION:

Get AppMenu Mobile Items

Gets a list of the App Menu items in the Salesforce mobile app for Android and iOS and the mobile web navigation menu. This resource is available in REST API version 29.0 and later.

Syntax

URI

`/services/data/vXX.X/appMenu/Salesforce1/`

Formats

JSON, XML

HTTP methods

GET

Authentication

`Authorization: Bearer token`

Request body

None

Request parameters

None required

Example

Example Request

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/appMenu/Salesforce1/
-H "Authorization: Bearer token"
```

Example Response Body


```
{
  "appMenuItems" : [ {
    "type" : "Standard.Search",
    "content" : null,
```

```
    "icons" : null,
    "colors" : null,
    "label" : "Smart Search Items",
    "url" : "/search"
  }, {
    "type" : "Standard.MyDay",
    "content" : null,
    "icons" : null,
    "colors" : null,
    "label" : "Today",
    "url" : "/myDay"
  }, {
    "type" : "Standard.Tasks",
    "content" : null,
    "icons" : null,
    "colors" : null,
    "label" : "Tasks",
    "url" : "/tasks"
  }, {
    "type" : "Standard.Dashboards",
    "content" : null,
    "icons" : null,
    "colors" : null,
    "label" : "Dashboards",
    "url" : "/dashboards"
  }, {
    "type" : "Tab.flexiPage",
    "content" : "MySampleFlexiPage",
    "icons" : [ {
      "contentType" : "image/png",
      "width" : 32,
      "height" : 32,
      "theme" : "theme3",
      "url" : "http://myorg.com/img/icon/custom51_100/bell32.png"
    }, {
      "contentType" : "image/png",
      "width" : 16,
      "height" : 16,
      "theme" : "theme3",
      "url" : "http://myorg.com/img/icon/custom51_100/bell16.png"
    }, {
      "contentType" : "image/svg+xml",
      "width" : 0,
      "height" : 0,
      "theme" : "theme4",
      "url" : "http://myorg.com/img/icon/t4/custom/custom53.svg"
    }, {
      "contentType" : "image/png",
      "width" : 60,
      "height" : 60,
      "theme" : "theme4",
      "url" : "http://myorg.com/img/icon/t4/custom/custom53_60.png"
    }, {
      "contentType" : "image/png",
```

```
      "width" : 120,
      "height" : 120,
      "theme" : "theme4",
      "url" : "http://myorg.com/img/icon/t4/custom/custom53_120.png"
    } ],
    "colors" : [ {
      "context" : "primary",
      "color" : "FC4F59",
      "theme" : "theme4"
    }, {
      "context" : "primary",
      "color" : "FC4F59",
      "theme" : "theme3"
    } ],
    "label" : "My App Home Page",
    "url" : "/servlet/servlet.Integration?lid=01rxx0000000Vsd&ic=1"
  }, {
    "type" : "Tab.apexPage",
    "content" : "/apex/myapexpage",
    "icons" : [ {
      "contentType" : "image/png",
      "width" : 32,
      "height" : 32,
      "theme" : "theme3",
      "url" : "http://myorg.com/img/icon/cash32.png"
    }, {
      "contentType" : "image/png",
      "width" : 16,
      "height" : 16,
      "theme" : "theme3",
      "url" : "http://myorg.com/img/icon/cash16.png"
    }, {
      "contentType" : "image/svg+xml",
      "width" : 0,
      "height" : 0,
      "theme" : "theme4",
      "url" : "http://myorg.com/img/icon/t4/custom/custom41.svg"
    }, {
      "contentType" : "image/png",
      "width" : 60,
      "height" : 60,
      "theme" : "theme4",
      "url" : "http://myorg.com/img/icon/t4/custom/custom41_60.png"
    }, {
      "contentType" : "image/png",
      "width" : 120,
      "height" : 120,
      "theme" : "theme4",
      "url" : "http://myorg.com/img/icon/t4/custom/custom41_120.png"
    } ],
    "colors" : [ {
      "context" : "primary",
      "color" : "3D8D8D",
      "theme" : "theme4"
```

```
    }, {
      "context" : "primary",
      "color" : "3D8D8D",
      "theme" : "theme3"
    } ],
    "label" : "label",
    "url" : "/servlet/servlet.Integration?lid=01rxx0000000Vyb&ic=1"
  } ]
}
```

Return Headers of AppMenu Mobile Item Requests

Returns only the headers that are returned by a GET request to the Salesforce mobile app for Android and iOS and the mobile web navigation menu. Use this URI to see the header values before you retrieve the content of the resource. This resource is available in REST API version 29.0 and later.

Syntax

URI

`/services/data/vXX.X/appMenu/Salesforce1/`

Formats

JSON, XML

HTTP methods

HEAD

Authentication

`Authorization: Bearer token`

Request body

None

Request parameters

None required

Example

Example Request

```
curl -X HEAD --head
https://MyDomainName.my.salesforce.com/services/data/v60.0/appMenu/Salesforce1 -H
"Authorization: Bearer token"
```

Compact Layouts

Returns a list of compact layouts for multiple objects. This resource is available in REST API version 31.0 and later.

This resource returns the primary compact layout for a set of objects. The set of objects is specified using a query parameter. Up to 100 objects can be queried at once.

Note: PersonAccount isn't supported for bulk queries. If you want to get the primary compact layout for PersonAccount, get it directly from `/services/data/v60.0/sobjects/Account/describe/compactLayouts/primaryPersonAccount`.

Syntax

URI

`/services/data/vXX.X/compactLayouts?q=objectList`

Formats

JSON, XML

HTTP methods

GET

Authentication

`Authorization: Bearer token`

Request parameters

Parameter	Description
q	A comma-delimited list of objects. The primary compact layout for each object in this list will be returned in the response of this resource.

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/compactLayouts?q=Account,Contact,CustomObj__c
 -H "Authorization: Bearer token"
```

Example Response Body


```
{
  "Account" : {
    "actions" : [ {
      "behavior" : null,
      "content" : null,
      "contentSource" : null,
      "custom" : false,
      "encoding" : null,
      "height" : null,
      "icons" : null,
      "label" : "Call",
      "menubar" : false,
      "name" : "CallHighlightAction",
      "overridden" : false,
      "resizeable" : false,
      "scrollbars" : false,
      "showsLocation" : false,
      "showsStatus" : false,
      "toolbar" : false,
```

```
      "url" : null,
      "width" : null,
      "windowPosition" : null
    },
    ...
    "id" : "0AHD000000000AbOAI",
    "label" : "Custom Account Compact Layout",
    "name" : "Custom_Account_Compact_Layout"
  },
  "Contact" : {
    "actions" : [ {
      "behavior" : null,
      "content" : null,
      "contentSource" : null,
      "custom" : false,
      "encoding" : null,
      "height" : null,
      "icons" : null,
      "label" : "Call",
      "menubar" : false,
      "name" : "CallHighlightAction",
      "overridden" : false,
      "resizeable" : false,
      "scrollbars" : false,
      "showsLocation" : false,
      "showsStatus" : false,
      "toolbar" : false,
      "url" : null,
      "width" : null,
      "windowPosition" : null
    },
    ...
    "id" : null,
    "label" : "System Default",
    "name" : "SYSTEM"
  }
  "CustomObj__c" : {
    "actions" : [ {
      "behavior" : null,
      "content" : null,
      "contentSource" : null,
      "custom" : false,
      "encoding" : null,
      "height" : null,
      "icons" : null,
      "label" : "Call",
      "menubar" : false,
      "name" : "CallHighlightAction",
      "overridden" : false,
      "resizeable" : false,
      "scrollbars" : false,
      "showsLocation" : false,
      "showsStatus" : false,
      "toolbar" : false,
```

```
      "url" : null,
      "width" : null,
      "windowPosition" : null
    },
    ...
    "id" : null,
    "imageItems" : null,
    "label" : "System Default",
    "name" : "SYSTEM"
  }
}
```

Consent

Your users can store consent preferences in different locations and possibly inconsistently. You can locate your customers' preferences for consent across multiple records when using API version 44.0 and later. Tracking consent preferences helps you and your users respect the most restrictive requests. You can use the Consent API with specific Data Cloud parameters with API version 50.0 and later.

Compile Consent Settings

Gets consent details based on a single action, like email or track, across specific consent management objects when the records have a lookup relationship. This resource is available in REST API version 45.0 and later.

To call Consent API, you must have either the View All Data or the Allow User Access to Privacy Data user permission. Requiring a perm ensures that the System Administrator gives explicit permission. This API accesses org-wide consent data, such as links between records and the value of consent flags, not just records to which the user ordinarily has access.

Consent API gets consent details based on a single action, like email or track, across the Contact, Contact Point Type Consent, Data Use Purpose, Individual, Lead, Person Account, and User objects when the records have a lookup relationship.

When you select *email* as the action, the API only aggregates consent for records that contain the same email address. If the record ID specified in the URI is associated with a record that contains a different email address, the consent settings of the associated record aren't included in the API response. The Consent API can't locate records in which the email address field is protected by Platform Encryption.

Note: When the API compares consent settings across records, it doesn't incorporate settings from converted leads.

Action	Fields Consulted	API Response	Response Schema

email	Contact.HasOptedOutOfEmail
	ContactPointTypeConsent.ContactPointType
	ContactPointTypeConsent.EffectiveFrom
	ContactPointTypeConsent.EffectiveTo
	ContactPointTypeConsent.PrivacyConsentStatus
	DataUsePurpose.Name
	Lead.HasOptedOutOfEmail
	PersonAccount.HasOptedOutOfEmail
		Within the time range if specified in ContactPointTypeConsent:
		Returns TRUE if all consulted field values are 0.
		Returns FALSE if any consulted field value is 1 or if no related Contact, Contact Point Type Consent, Lead, or
			{
			"<ID/Email>" :
			{
			"result" : "<Success/errormessage>",
			"proceed" : { "emailResult" : "<Success/errormessage>", email : "<true/false>" }
			}
			}

		Person Account object exists.	

fax	Contact.HasOptedOutOfFax
	DataUsePurpose.Name
	Lead.HasOptedOutOfFax
	PersonAccount.HasOptedOutOfFax
		Returns TRUE if all consulted field values are 0.
		Returns FALSE if any consulted field value is 1 or if no related Contact, Lead, or Person Account object exists.
			{
			"<ID/Email>" :
			{
			"result" : "<Success/errormessage>",
			"proceed" : { "faxResult" : "<Success/errormessage>", fax : "<true/false>" }
			}
			}

geotrack	DataUsePurpose.Name
	Individual.HasOptedOutGeoTracking
		Returns TRUE if the consulted field value is 0.
		Returns FALSE if the consulted field value is 1 or if no related Individual object exists.
			{
			"<ID/Email>" :
			{
			"result" : "<Success/errormessage>",
			"proceed" : { "geotrackResult" : "<Success/errormessage>", "geotrack" : "<true/false>" }
			}
			}

mail	ContactPointTypeConsent.ContactPointType
	ContactPointTypeConsent.EffectiveFrom
	ContactPointTypeConsent.EffectiveTo
	ContactPointTypeConsent.PrivacyConsentStatus
	DataUsePurpose.Name
		Within the time range if specified in ContactPointTypeConsent:
		Returns TRUE if all consulted field values are 0.
		Returns FALSE if any consulted field value is 1 or if no related Contact, Contact Point Type Consent, Lead, or Person Account object exists.
			{
			"<ID/Email>" :
			{
			"result" : "<Success/errormessage>",
			"proceed" : { "mailingResult" : "<Success/errormessage>", "mail" : "<true/false>" }
			}
			}

phone	Contact.DoNotCall ContactPointTypeConsent.ContactPointType ContactPointTypeConsent.EffectiveFrom ContactPointTypeConsent.EffectiveTo ContactPointTypeConsent.PrivacyConsentStatus	Within the time range if specified in ContactPointTypeConsent: Returns TRUE if all consulted field values are 0.	{ "<ID/Email>" : { "result" : "<Success/errormessage>", "proceed" : { "phoneResult" : "<Success/errormessage>", "phone" : "<true/false>" }

	DataUsePurpose.Name Lead.DoNotCall PersonAccount.DoNotCall	Returns FALSE if any consulted field value is 1 or if no related Contact, Contact Point Type Consent, Lead, or Person Account object exists.	} }

portability	DataUsePurpose.Name
	Individual.SendIndividualData
		Returns TRUE if the consulted field value is 1.
		Returns FALSE if the consulted field value is 0 or if no related Individual object exists.
			{
			"<ID/Email>" :
			{
			"result" : "<Success/errormessage>",
			"proceed" : { "portabilityResult" : "<Success/errormessage>", "portability" : "<true/false>"
			}
			}
			}

process	DataUsePurpose.Name
	Individual.HasOptedOutProcessing
		Returns TRUE if the consulted field value is 0.
		Returns FALSE if the consulted field value is 1 or if no related Individual object exists.
			{
			"<ID/Email>" :
			{
			"result" : "<Success/errormessage>",
			"proceed" : { "processResult" : "<Success/errormessage>", "process" : "<true/false>" }
			}
			}

profile	DataUsePurpose.Name
	Individual.HasOptedOutProfiling
		Returns TRUE if the consulted field value is 0.
		Returns FALSE if the consulted field value is 1 or if no related Individual object exists.
			{
			"<ID/Email>" :
			{
			"result" : "<Success/errormessage>",
			"proceed" : { "profileResult" : "<Success/errormessage>", "profile" : "<true/false>" }
			}
			}

shouldforget	DataUsePurpose.Name Individual.ShouldForget	Returns TRUE if the consulted field value is 1. Returns FALSE if the consulted field value is	{ "<ID/Email>" : { "result" : "<Success/errormessage>",

		0 or if no related Individual object exists.	"proceed" : { "shouldForgetResult" : "<Success/errormessage>", "shouldforget" : "<true/false>" } } }

social	ContactPointTypeConsent.ContactPointType
	ContactPointTypeConsent.EffectiveFrom
	ContactPointTypeConsent.EffectiveTo
	ContactPointTypeConsent.PrivacyConsentStatus
	DataUsePurpose.Name
		Within the time range if specified in ContactPointTypeConsent:
		Returns TRUE if all consulted field values are 0.
		Returns FALSE if any consulted field value is 1 or if no related Contact, Contact Point Type Consent, Lead, or Person Account object exists.
			{
			"<ID/Email>" :
			{
			"result" : "<Success/errormessage>",
			"proceed" : { "socialResult" : "<Success/errormessage>", "social" : "<true/false>" }
			}
			}

solicit	DataUsePurpose.Name
	Individual.HasOptedOutSolicit
		Returns TRUE if the consulted field value is 0.
		Returns FALSE if the consulted field value is 1 or if no related Individual object exists.
			{
			"<ID/Email>" :
			{
			"result" : "<Success/errormessage>",
			"proceed" : { "solicitResult" : "<Success/errormessage>", "solicit" : "<true/false>" }
			}
			}

storepiielsewhere	DataUsePurpose.Name
	Individual.CanStorePiiElsewhere
		Returns TRUE if the consulted field value is 1.
		Returns FALSE if the consulted field value is 0 or if no related Individual object exists.
			{
			"<ID/Email>" :
			{
			"result" : "<Success/errormessage>",
			"proceed" : { "storePIIElsewhereResult" : "<Success/errormessage>", "storepiielsewhere" : "<true/false>" }
			}
			}

track	DataUsePurpose.Name Individual.HasOptedOutTracking	Returns TRUE if the consulted field value is 0.	{ "<ID/Email>" : {

		Returns FALSE if the consulted field value is 1 or if no related Individual object exists.	"result" : "<Success/errormessage>", "proceed" : { "trackResult" : "<Success/errormessage>", "track" : "<true/false>" } } }

web	ContactPointTypeConsent.ContactPointType
	ContactPointTypeConsent.EffectiveFrom
	ContactPointTypeConsent.EffectiveTo
	ContactPointTypeConsent.PrivacyConsentStatus
	DataUsePurpose.Name
		Within the time range if specified in ContactPointTypeConsent.
		Returns TRUE if all consulted field values are 0.
		Returns FALSE if any consulted field value is 1 or if no related Contact, Contact Point Type Consent, Lead, or Person Account object exists.
			{
			"<ID/Email>" :
			{
			"result" : "<Success/errormessage>",
			"proceed" : { "webResult" : "<Success/errormessage>", "web" : "<true/false>" }
			}
			}

Syntax

URI

`/services/data/vXX.X/consent/action/action?ids=listOfIds`

Formats

JSON

HTTP methods

GET

Authentication

`Authorization: Bearer token`

Request body

None

Request parameters

Parameter	Description
`action`	Required. The proposed action. If this parameter is used, `actions` can't be used.
`aggregatedConsent`	Optional: true or false. `aggregatedConsent` is the same as `aggregatedConsent=true`. If true, one result is returned indicating whether to proceed or not, rather than a result for each ID. If any ID in the list returns false, the aggregated result is false.

Parameter	Description
`datetime`	Optional. The timestamp for which consent is determined. The value is converted to the UTC timezone and must be formatted as described in Valid Date and DateTime Formats. If not specified, defaults to the current date and time.
`ids`	Required. Comma-separated list of IDs. The ID can be the record ID or the email address listed on the record.
`policy`	Optional. Use `policy=requireExplicitConsent` to specify in the API response whether explicit consent was given for a contact point channel. The API returns an infoNotFound response when consent isn't specified. This parameter is available in API version 49.0 and later.
`purpose`	Optional. The reason for contacting a customer.
`verbose`	Optional: true or false. `verbose` is the same as `verbose=true`. Verbose responses are slower than non-verbose responses. See the examples for a verbose response.

Example

Request for URI structure

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/consent/action/track?ids=003xx000004TxyY,00Qxx00000syyO,003zz000004zzZ
 -H "Authorization: Bearer token"
```

Request for Email addresses as IDs, specified purpose and timespan, and a verbose response

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/consent/action/email?ids=j0t5t5b2@tkbxp5ia.com,4qxlswo@23wj7pwh.com&datetime=2018-12-12T00:00:00Z
 -H "Authorization: Bearer token"
```

Response Body

```
{
  "j0t5t5b2@tkbxp5ia.com" : {
    "result" : "Success",
    "proceed" : {
      "email" : "true"
      "emailResult" : "Success"
    },
    "explanation" : [ {
      "objectConsulted" : "ContactTypePointConsent",
      "status" : "opt_in",
      "purpose" : "billing",
      "recordId" : "003xx000004TxyY",
      "value" : "true"
    },{
      "objectConsulted" : "Contact",
      "field" : "HasOptedOutOfTracking",
      "recordId" : "1",
      "value" : "true"
    }]
```

```
  },
  "4quxlswo@23wj7pwh.com" : {
    "result" : "Success",
    "proceed" : {
      "email" : "false"
      "emailResult" : "Success"
    },
    "explanation" : [ {
      "objectConsulted" : "Contact",
      "field" : "HasOptedOutOfEmail",
      "recordId" : "00Qxx00000skwO",
      "value" : "true"
    }  ]
  }
}
```

Compile Multiple Types of Consent Settings

Gets consent details based on multiple actions, like email and track, across specific consent management objects when the records have a lookup relationship. Available in API version 45.0 and later.

To call Consent API, you must have either the View All Data or the Allow User Access to Privacy Data user permission. Requiring a perm ensures that the System Administrator gives explicit permission. This API accesses org-wide consent data, such as links between records and the value of consent flags, not just records to which the user ordinarily has access.

Consent API gets consent details across the Contact, Contact Point Type Consent, Data Use Purpose, Individual, Lead, Person Account, and User objects when the records have a lookup relationship.

The following table shows how the API responses are determined. If the consulted fields find conflicting consent preferences, the response returns the least permissive preference. For example, if Contact.HasOptedOutOfEmail is `false`, but Lead.HasOptedOutOfEmail is `true`, the response indicates that you can't proceed with emailing the user.

When you select *email* as the action, the API only aggregates consent for records that contain the same email address. If the record ID specified in the URI is associated with a record that contains a different email address, the consent settings of the associated record aren't included in the API response.

Note: When the API compares consent settings across records, it doesn't incorporate settings from converted leads.

Action	Fields Consulted	API Response	Response Schema

email	Contact.HasOptedOutOfEmail
	ContactPointTypeConsent.ContactPointType
	ContactPointTypeConsent.EffectiveFrom
	ContactPointTypeConsent.EffectiveTo
	ContactPointTypeConsent.PrivacyConsentStatus
	DataUsePurpose.Name
	Lead.HasOptedOutOfEmail
	PersonAccount.HasOptedOutOfEmail
		Within the time range if specified in ContactPointTypeConsent:
		Returns TRUE if all consulted field values are 0.
		Returns FALSE if any consulted field value is 1 or if no related Contact, Contact Point Type Consent, Lead, or
			{
			"<ID/Email>" :
			{
			"result" : "<Success/errormessage>",
			"proceed" : { "emailResult" : "<Success/errormessage>", email : "<true/false>" }
			}
			}

		Person Account object exists.	

fax	Contact.HasOptedOutOfFax
	DataUsePurpose.Name
	Lead.HasOptedOutOfFax
	PersonAccount.HasOptedOutOfFax
		Returns TRUE if all consulted field values are 0.
		Returns FALSE if any consulted field value is 1 or if no related Contact, Lead, or Person Account object exists.
			{
			"<ID/Email>" :
			{
			"result" : "<Success/errormessage>",
			"proceed" : { "faxResult" : "<Success/errormessage>", fax : "<true/false>" }
			}
			}

geotrack	DataUsePurpose.Name
	Individual.HasOptedOutGeoTracking
		Returns TRUE if the consulted field value is 0.
		Returns FALSE if the consulted field value is 1 or if no related Individual object exists.
			{
			"<ID/Email>" :
			{
			"result" : "<Success/errormessage>",
			"proceed" : { "geotrackResult" : "<Success/errormessage>", "geotrack" : "<true/false>" }
			}
			}

mail	ContactPointTypeConsent.ContactPointType
	ContactPointTypeConsent.EffectiveFrom
	ContactPointTypeConsent.EffectiveTo
	ContactPointTypeConsent.PrivacyConsentStatus
	DataUsePurpose.Name
		Within the time range if specified in ContactPointTypeConsent:
		Returns TRUE if all consulted field values are 0.
		Returns FALSE if any consulted field value is 1 or if no related Contact, Contact Point Type Consent, Lead, or Person Account object exists.
			{
			"<ID/Email>" :
			{
			"result" : "<Success/errormessage>",
			"proceed" : { "mailingResult" : "<Success/errormessage>", "mail" : "<true/false>" }
			}
			}

phone	Contact.DoNotCall ContactPointTypeConsent.ContactPointType ContactPointTypeConsent.EffectiveFrom ContactPointTypeConsent.EffectiveTo ContactPointTypeConsent.PrivacyConsentStatus	Within the time range if specified in ContactPointTypeConsent: Returns TRUE if all consulted field values are 0.	{ "<ID/Email>" : { "result" : "<Success/errormessage>", "proceed" : { "phoneResult" : "<Success/errormessage>", "phone" : "<true/false>" }

	DataUsePurpose.Name Lead.DoNotCall PersonAccount.DoNotCall	Returns FALSE if any consulted field value is 1 or if no related Contact, Contact Point Type Consent, Lead, or Person Account object exists.	} }

portability	DataUsePurpose.Name
	Individual.SendIndividualData
		Returns TRUE if the consulted field value is 1.
		Returns FALSE if the consulted field value is 0 or if no related Individual object exists.
			{
			"<ID/Email>" :
			{
			"result" : "<Success/errormessage>",
			"proceed" : { "portabilityResult" : "<Success/errormessage>", "portability" : "<true/false>"
			}
			}
			}

process	DataUsePurpose.Name
	Individual.HasOptedOutProcessing
		Returns TRUE if the consulted field value is 0.
		Returns FALSE if the consulted field value is 1 or if no related Individual object exists.
			{
			"<ID/Email>" :
			{
			"result" : "<Success/errormessage>",
			"proceed" : { "processResult" : "<Success/errormessage>", "process" : "<true/false>" }
			}
			}

profile	DataUsePurpose.Name
	Individual.HasOptedOutProfiling
		Returns TRUE if the consulted field value is 0.
		Returns FALSE if the consulted field value is 1 or if no related Individual object exists.
			{
			"<ID/Email>" :
			{
			"result" : "<Success/errormessage>",
			"proceed" : { "profileResult" : "<Success/errormessage>", "profile" : "<true/false>" }
			}
			}

shouldforget	DataUsePurpose.Name Individual.ShouldForget	Returns TRUE if the consulted field value is 1. Returns FALSE if the consulted field value is	{ "<ID/Email>" : { "result" : "<Success/errormessage>",

		0 or if no related Individual object exists.	"proceed" : { "shouldForgetResult" : "<Success/errormessage>", "shouldforget" : "<true/false>" } } }

social	ContactPointTypeConsent.ContactPointType
	ContactPointTypeConsent.EffectiveFrom
	ContactPointTypeConsent.EffectiveTo
	ContactPointTypeConsent.PrivacyConsentStatus
	DataUsePurpose.Name
		Within the time range if specified in ContactPointTypeConsent:
		Returns TRUE if all consulted field values are 0.
		Returns FALSE if any consulted field value is 1 or if no related Contact, Contact Point Type Consent, Lead, or Person Account object exists.
			{
			"<ID/Email>" :
			{
			"result" : "<Success/errormessage>",
			"proceed" : { "socialResult" : "<Success/errormessage>", "social" : "<true/false>" }
			}
			}

solicit	DataUsePurpose.Name
	Individual.HasOptedOutSolicit
		Returns TRUE if the consulted field value is 0.
		Returns FALSE if the consulted field value is 1 or if no related Individual object exists.
			{
			"<ID/Email>" :
			{
			"result" : "<Success/errormessage>",
			"proceed" : { "solicitResult" : "<Success/errormessage>", "solicit" : "<true/false>" }
			}
			}

storepiielsewhere	DataUsePurpose.Name
	Individual.CanStorePiiElsewhere
		Returns TRUE if the consulted field value is 1.
		Returns FALSE if the consulted field value is 0 or if no related Individual object exists.
			{
			"<ID/Email>" :
			{
			"result" : "<Success/errormessage>",
			"proceed" : { "storePIIElsewhereResult" : "<Success/errormessage>", "storepiielsewhere" : "<true/false>" }
			}
			}

track	DataUsePurpose.Name Individual.HasOptedOutTracking	Returns TRUE if the consulted field value is 0.	{ "<ID/Email>" : {

		Returns FALSE if the consulted field value is 1 or if no related Individual object exists.	"result" : "<Success/errormessage>", "proceed" : { "trackResult" : "<Success/errormessage>", "track" : "<true/false>" } } }

web	ContactPointTypeConsent.ContactPointType
	ContactPointTypeConsent.EffectiveFrom
	ContactPointTypeConsent.EffectiveTo
	ContactPointTypeConsent.PrivacyConsentStatus
	DataUsePurpose.Name
		Within the time range if specified in ContactPointTypeConsent:
		Returns TRUE if all consulted field values are 0.
		Returns FALSE if any consulted field value is 1 or if no related Contact, Contact Point Type Consent, Lead, or Person Account object exists.
			{
			"<ID/Email>" :
			{
			"result" : "<Success/errormessage>",
			"proceed" : { "webResult" : "<Success/errormessage>", "web" : "<true/false>" }
			}
			}

Syntax

URI

/services/data/v***XX.X***/consent/multiaction?actions=***listOfActions***&ids=***listOfIds***

Formats

JSON

HTTP methods

GET

Authentication

Authorization: Bearer ***token***

Request body

None

Request parameters

Parameter	Description
actions	Required. Comma-separated list of proposed actions. If this parameter is used, `action` can't be used.
aggregatedConsent	Optional: true or false. `aggregatedConsent` is the same as `aggregatedConsent=true`. If true, one result is returned indicating whether to proceed or not, rather than a result for each ID. If any ID in the list returns false, the aggregated result is false.

Parameter	Description
`datetime`	Optional. The timestamp for which consent is determined. The value is converted to the UTC timezone and must be formatted as described in Valid Date and DateTime Formats. If not specified, defaults to the current date and time.
`ids`	Required. Comma-separated list of IDs. The ID can be the record ID or the email address listed on the record.
`policy`	Optional. Use `policy=requireExplicitConsent` to specify in the API response whether explicit consent was given for a contact point channel. The API returns an infoNotFound response when consent isn't specified. This parameter is available in API version 49.0 and later.
`purpose`	Optional. The reason for contacting a customer.
`verbose`	Optional: true or false. `verbose` is the same as `verbose=true`. Verbose responses are slower than non-verbose responses. See the examples for a verbose response.

Example

Request for Multiaction URI structure

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/consent/multiaction?actions=track,geotrack,email&ids=003xx000008TiyY,00Qxx00000skwO,dek65@tf7h.com
 -H "Authorization: Bearer token"
```

Request for email addresses as IDs, specified purpose and timespan, and a verbose response

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/consent/action/email?ids=j0t5t5b2@tkbxp5ia.com,4pxlsw@23wj7pvh.com&datetime=2018-12-12T00:00:00Z&purpose=billing&verbose=true
 -H "Authorization: Bearer token"
```

Response Body

```
{
  "j0t5t5b2@tkbxp5ia.com" : {
    "result" : "Success",
    "proceed" : {
       "email" : "false"
      "emailResult" : "Success"
      "track" : "false"
      "trackResult" : "Success"
      "solicit" : "false"
      "solicitResult" : "Success"
    },
    "explanation" : [ {
      "objectConsulted" : "ContactTypePointConsent",
      "status" : "opt_in",
      "purpose" : "billing",
      "recordId" : "003xx000004TxyY",
      "value" : "true"
    },{
      "objectConsulted" : "Individual",
```

```
        "field" : "HasOptedOutOfTracking",
        "recordId" : "0PKx000006JkyZ",
        "value" : "true"
      }]
    },
    "4quxlswo@23wj7pwh.com" : {
      "result" : "Success",
      "proceed" : {
          "email" : "false"
        "emailResult" : "Success"
        "track" : "false"
        "trackResult" : "Success"
        "solicit" : "true"
        "solicitResult" : "Success"
      },
      "explanation" : [ {
        "objectConsulted" : "Contact",
        "field" : "HasOptedOutOfEmail",
        "recordId" : "00Qxx00000skwO",
        "value" : "true"
      },{
        "objectConsulted" : "Individual",
        "field" : "HasOptedOutOfSolicit",
        "recordId" : "0PKx000003JcpK",
        "value" : "false"
      }]
    }
  }
```

Use the Consent API with Data Cloud

The Consent API supports Data Cloud. Use the Consent API to read and write to the Data Cloud profile. Contact your Salesforce Representative for consumer rights guidance within Data Cloud.

Required Permissions

To use Data Cloud parameters for Consent API, you must have either the ModifyAllData or the ConsentApiUpdate user permission. Requiring a perm ensures that the Salesforce admin gives explicit permission. These parameters write org-wide consent data, such as links between records and the value of consent flags, which are usually inaccessible to non-admin users.

Actions Supported by Consent API with Data Cloud

Action	Description
Processing	This action is used to restrict processing of data in Data Cloud processes such as query and segmentation.
Portability	This action is used to allow export of Data Cloud profile data.
Shouldforget	This action indicates the right to be forgotten, which means delete PII (Personally Identifiable Information) data and any related records.

Data Cloud Read Parameters

The Consent API allows you to gather information about the Data Cloud profile. Use the `mode` and `ids` Data Cloud parameters as described below.

Syntax

HTTP method

GET

Available since release:

48.0

URI

Note: You can access the consent API using three different URIs based on the Action. The Actions supported are `processing`,`portability`, and `shouldforget`.

```
/services/data/vXX.X/consent/action/processing?ids=<list_of_ids>&mode=<cdp>
/services/data/vXX.X/consent/action/portability?ids=<list_of_ids>&mode=<cdp>
/services/data/vXX.X/consent/action/shouldforget?ids=<list_of_ids>&mode=<cdp>
```

Request parameters

Parameter	Description
`ids`	Required. Comma-separated list of IDs. The ID provided must be present in a field mapped to Individual.Individual Id.
`mode`	Optional. Default is normal. Valid value to retrieve a Data Cloud profile is `cdp`.

Read Example

URI

```
/services/data/v60.0/consent/action/portability?ids=00932I3SU92&mode=cdp
```

Response

```
{ "j00932I3SU92" : { "result" : "Success", "proceed" : { "portability" : "true"
"portabilityResult" : "Success" } } }
```

Write Parameters

The Consent API also allows you to write information to the Data Cloud profile. Use the ids, mode, and status parameters as described below.

Note: You can update your consent information with the consent API using three different URIs. The URIs are based on the action that is to be performed on the Data Cloud profile. The actions supported are `processing`, `portability`, and `shouldforget`.

Syntax

HTTP method

PATCH

Available since release

50.0

URI when action is processing

`/services/data/vXX.X/consent/action/processing?ids=list_of_ids&mode=cdp&status=optin or optout`

Request parameters when action is processing

Parameter	Description
`ids`	Required. Comma-separated list of IDs. The ID provided must be present in a field mapped to Individual.Individual Id.
`mode`	Optional. Default is normal. Valid value to use for updating a Data Cloud profile is `cdp`.
`status`	Required. Status of the consent. Allowed values are `optin` or `optout`. However, when action is processing use status as `optout`.

URI when action is shouldforget

`/services/data/vXX.X/consent/action/shouldforget?ids=list_of_ids&mode=cdp&status=optin or optout`

Request parameters when action is shouldforget

Parameter	Description
`ids`	Required. Comma-separated list of IDs. The ID provided must be present in a field mapped to Individual.Individual Id.
`mode`	Optional. Default is normal. Valid value to use for updating a Data CloudCDP profile is `cdp`.
`status`	Required. Status of the consent. Allowed values are `optin` or `optout`. However, when action is shouldforget use status as `optin`.

URI action is portability

`/services/data/vXX.X/consent/action/portability?ids=list_of_ids&mode=cdp&status=optin or optout`

Request parameters when action is portability

Parameter	Description
`ids`	Required. Comma-separated list of IDs. The ID provided must be present in a field mapped to Individual.Individual Id.
`mode`	Optional. Default is normal. Valid value to use for updating a Data Cloud profile is `cdp`.
`status`	Required. Status of the consent. Allowed values are `optin` or `optout`. When action is portability use status as `optin`.

Parameter	Description
aws_s3_bucket_id	Required only when mode is 'cdp' and the action is 'portability'. This parameter must be passed in as part of the PATCH request body. This parameter is used to pass the S3 bucket location for portability requests to Data Cloud.
aws_access_key_id	Required only when mode is 'cdp' and the action is 'portability'. This parameter must be passed in as part of the PATCH request body. This parameter is used to pass the S3 bucket access key for portability requests to Data Cloud.
aws_secret_access_key	Required only when mode is 'cdp' and the action is 'portability'. This parameter must be passed in as part of the PATCH request body. This parameter is used to pass the S3 bucket secret access key for portability requests to Data Cloud.
aws_s3_folder	Required only when mode is 'cdp' and the action is 'portability'. This parameter must be passed in as part of the PATCH request body. This parameter is used to pass the S3 bucket folder for portability requests to Data Cloud.
aws_region	Required only when mode is 'cdp' and the action is 'portability'. This parameter must be passed in as part of the PATCH request body. This parameter is used to pass the S3 bucket's aws region for portability requests to Data Cloud.

Write Example

When action is processing

```
/services/data/v60.0/consent/action/processing?ids=100000695&mode=cdp&status=optout
body: {}
```

When action is portability

```
/services/data/v60.0/consent/action/portability?ids=100000695&mode=cdp&status=optin
   body:{
          "aws_s3_bucket_id" : "cdpgdprtest",
          "aws_access_key_id": "ABCD1234WEXAMPLE",
          "aws_secret_access_key": "WXYZ1234EXAMPLE",
          "aws_s3_folder": "yyun/Person",
          "aws_region": "us-west-1"
        }
```

When action is shouldforget

```
/services/data/v60.0/consent/action/shouldforget?ids=100000695 &mode=cdp&status=optin
body: {}
```

Consent Write

Your users can store consent preferences in different locations. The Consent Write API can update and write consent across multiple records through a single API call, helping you sync consent across records or populate the new Consent data model. This resource is available in REST API version 48.0 and later.

Consent API writes consent values across the Contact, Contact Point Type Consent, Data Use Purpose, Individual, Lead, Person Account, and User objects when the records have a lookup relationship or share an email address. This API can also write to the Data Cloud Individual record. The Consent API can't locate records in which the email address field is protected by Platform Encryption.

Note: For the Spring '21 release, the API only takes in a single email address. Any record with a matching email address is updated based on the parameters set in the API call.

All records with the email address listed are updated. If the Create Individual parameter is selected and no Individual record exists, the API creates an Individual record. If warranted, the API also creates a Contact Point Type Consent and Contact Point Email record.

Only Data Cloud uses the request body. If not passing anything in the request body, pass in an empty object {}.

Syntax

URI

`/services/data/vXX.X/consent/action/action?ids=listOfIds`

Formats

JSON

HTTP methods

PATCH

Authentication

`Authorization: Bearer token`

Request parameters

Parameter	Description
`blobParam`	Optional. Use to pass information to Data Cloud, such as portability write location. Use only for `mode=cdp`. This parameter must be passed in as part of the PATCH request body.
`captureDate`	Optional. The date and time when consent is captured. The default is the date and time the API call is made.
`captureContactPointType`	Optional. Describes how consent is captured (web, phone, email). Supported values are: • `email` • `phone` • `web` (default)
`captureSource`	Optional. The source through which consent is captured. The default capture source is Consent API. Max length 255 characters.
`consentName`	Optional. Use to set the name for any new consent records. Default is: `Individual Name-Datetime (<Name> 2019-03-31T15:47:57).` Max length is 255 characters.
`createIndividual`	Optional. Boolean. If set to `true` and the API call includes an email address that matches multiple records without an Individual object, then an Individual object is created. Any consent records with an email address that match the email in the API call are linked to the new Individual object. If multiple records are found, then any

Parameter	Description
	records not linked to an Individual object is linked to the Individual object found in the other records. If more than one Individual object is found on the matching records, then the call is rejected.
`doubleOptIn`	Optional. The date and time that the double opt-in is completed, formatted as described in Valid Date and DateTime Formats.
`effectiveFrom`	Optional. The date from which consent is effective, formatted as described in Valid Date and DateTime Formats. The default is the date the API call is made.
`effectiveTo`	Optional. The date through which consent is effective, formatted as described in Valid Date and DateTime Formats.
`ids`	Required. The email address used to sync consent. The ID can be the record ID or the email address listed on the record. When `mode=cdp`, the ID value is a string equal to the Individual ID attribute.
`individualName`	Optional. The name of the person in an Individual record. If a name isn't provided for a new Individual record, then the local part of the passed-in email address is used. Max length is 80 characters.
`mode`	Optional. Default is `normal`. The allowed modes are: `normal` or `cdp`. With `mode=cdp`, the request is passed to the Data Cloud platform to get or write consent. The `mode=cdp` parameter only supports the `action`, `blobParam`, and `ids` parameters.
`purposeName`	Optional. The data use purpose for which consent is provided. Must use an existing data use purpose that you previously created. If more than one purpose with the same name exists, only one purpose is selected.
`status`	Required. Status of the consent (`OptIn`, `OptOut`, `Seen`, `NotSeen`.) If action exists on an Individual object (for example, tracking or processing), the only valid values are `OptIn` and `OptOut`.

Action

Allowed values are:

- email
- fax
- geotrack
- mailing
- phone
- portability
- process
- profile
- shouldForget
- social
- solicit

- storePiiElsewhere
- track
- web

Security

To call the Consent Write API, you must have either the ModifyAllData or the ConsentApiUpdate user permission. This API writes org-wide consent data, such as links between records and the value of consent flags, and not just records to which the user ordinarily has access. The ConsentApiUpdate user permission grants full write permission to the user during the Consent Write API call.

Example

Example Request

```
curl -X PATCH
https://MyDomainName.my.salesforce.com/services/data/v60.0/consent/action/<action>?ids=<email-OR-recordID>&status=<optout/optin/seen/notseen>&createIndividual=<true/false>
 -H "Content-Type: application/json" -d "@exampleRequestBody.json"
```

Example Request Body

```
{}
```

Example Response Body

```
{
  "<email-OR-recordID>" : {
    "result" : "Success",
    "edited" : [{
      "objectType" : "<Contact, Lead, User, etc.>",
      "field" : "<HasOptedOutofFax, DoNotCall,etc>",
      "valueOfField" : "<true/false>",
      "id" : "<recordID>"
    }],
  }
}
```

Embedded Service Configuration Describe

Retrieves the values for your Embedded Service deployment configuration or the headers returned by a request.

IN THIS SECTION:

Get Embedded Service Configuration

Get the values for your Embedded Service deployment configuration, including the branding colors, font, and site URL. This resource is available in REST API version 45.0 and later.

Return Headers for Embedded Service Configuration

Returns only the headers from a GET request to the Embedded Service Configuration Describe resource. This gives you a chance to see header values ahead of time before retrieving the content of the resource. You must be logged in to the account that owns the EmbeddedServiceConfigDeveloperName you are querying. This resource is available in REST API version 45.0 and later.

Get Embedded Service Configuration

Get the values for your Embedded Service deployment configuration, including the branding colors, font, and site URL. This resource is available in REST API version 45.0 and later.

You must be logged in to the account that owns the EmbeddedServiceConfigDeveloperName you are querying.

Syntax

URI

`/services/data/vXX.X/support/embeddedservice/configuration/embeddedServiceConfigDeveloperName`

Formats

JSON

HTTP method

GET

Authentication

`Authorization: Bearer token`

Request parameters

None

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/support/embeddedservice/configuration/TestOne
 -H "Authorization: Bearer token"
```

Example Response Body

```
{
  "embeddedServiceConfig" : {
    "areGuestUsersAllowed" : false,
    "authMethod" : "CustomLogin",
    "embeddedServiceBranding" : {
      "contrastInvertedColor" : "#ffffff",
      "contrastPrimaryColor" : "#333333",
      "font" : "Salesforce Sans",
      "height" : 498,
      "navBarColor" : "#222222",
      "primaryColor" : "#222222",
      "secondaryColor" : "#005290",
      "width" : 320
    },
    "embeddedServiceLiveAgent" : {
      "avatarImg" : "",
      "embeddedServiceQuickActions" : [ {
        "order" : 1,
       "quickActionDefinition" : "Snapins_Case_OfflineCaseQuickAction_08hRM00000000cC",

        "quickActionType" : "OfflineCase"
      }, {
```

```
          "order" : 1,
         "quickActionDefinition" : "Snapins_Contact_PrechatQuickAction_08hRM00000000RC",

          "quickActionType" : "Prechat"
        }, {
          "order" : 2,
          "quickActionDefinition" : "Snapins_Case_PrechatQuickAction_08hRM00000000RC",
          "quickActionType" : "Prechat"
        } ],
        "enabled" : true,
        "fontSize" : "Medium",
        "headerBackgroundImg" : "https://google.com/img/headerBgImgUrl.png",
        "isOfflineCaseEnabled" : true,
        "isQueuePositionEnabled" : true,
        "liveChatButton" : "573RM0000004GGf",
        "liveChatDeployment" : "572RM0000004CDV",
        "offlineCaseBackgroundImg" : "https://google.com/img/offlineBgImgUrl.png",
        "prechatBackgroundImg" : "https://google.com/img/prechatBgImgUrl.png",
        "prechatEnabled" : true,
        "scenario" : "Service",
        "smallCompanyLogoImg" : "https://google.com/img/logoImgUrl.png",
        "waitingStateBackgroundImg" : "https://google.com/img/bgImgUrl.png"

      },
      "shouldHideAuthDialog" : false,
      "siteUrl" : "https://snapins-15f082fb956-15fbc261d27.stmfa.stm.force.com/napili2"
    }
}
```

Return Headers for Embedded Service Configuration

Returns only the headers from a GET request to the Embedded Service Configuration Describe resource. This gives you a chance to see header values ahead of time before retrieving the content of the resource. You must be logged in to the account that owns the EmbeddedServiceConfigDeveloperName you are querying. This resource is available in REST API version 45.0 and later.

Syntax

URI

`/services/data/vXX.X/support/embeddedservice/configuration/embeddedServiceConfigDeveloperName`

Formats

JSON

HTTP method

HEAD

Authentication

`Authorization: Bearer token`

Request parameters

None

Invocable Actions

Represents standard and custom invocable 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.

IN THIS SECTION:

SEE ALSO:

Apex Developer Guide : InvocableMethod Annotation

Get Invocable Actions

Gets standard and custom invocable action URIs from Salesforce. This resource is available in REST API version 32.0 and later.

Example

URI

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

Formats

JSON, XML

HTTP Methods

GET

Authentication

`Authorization: Bearer token`

Request parameters

None required

Example

Example Request

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/actions -H "Authorization:
 Bearer token"
```

Example Response Body

```
{
  "standard" : "/services/data/v60.0/actions/standard",
  "custom" : "/services/data/v60.0/actions/custom"
}
```

Return HTTP Headers for Invocable Actions

Returns only the headers that are returned by sending a GET request to the invocable actions resource. This gives you a chance to see returned header values of the GET request before retrieving the content. This resource is available in REST API version 32.0 and later.

URI

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

Formats

JSON, XML

HTTP Methods

HEAD

Authentication

`Authorization: Bearer token`

Request parameters

None required

Example

Example Request

```
curl -X HEAD --head https://MyDomainName.my.salesforce.com/services/data/v60.0/actions -H
 "Authorization: Bearer token"
```

Example Response Body

```
HTTP/1.1 200 OK
Date: Mon, 21 Nov 2022 22:56:26 GMT
```

Invocable Actions Custom

Represents custom invocable actions that can be statically invoked. You can also get basic information for each type of action.

IN THIS SECTION:

SEE ALSO:

Apex Developer Guide : InvocableMethod Annotation

Get Custom Invocable Actions

Gets the list of all custom invocable actions. Some actions require special access. This resource is available in REST API version 32.0 and later.

Sending email with the `emailAlert` action counts against your daily email limit for workflows. For more information, see Daily Allocations for Email Alerts in Salesforce Help.

When invoking an Apex action using the POST method and supplying the inputs in the request, only the following primitive types are supported as inputs:

- `Blob`
- `Boolean`
- `Date`
- `Datetime`
- `Decimal`
- `Double`
- `ID`
- `Integer`
- `Long`
- `String`
- `Time`

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 Invocable Method annotation from the Apex class, a runtime error in the flow occurs.

When a flow user invokes an autolaunched flow, the active flow version runs. If there's no active version, the latest version runs. When a flow admin invokes a flow, the latest version always runs.

If any of these elements are used in a flow, packageable components that reference the elements aren't automatically included in the package.

- Apex action
- Email alerts
- Post to Chatter core action
- Quick Action core action
- Send Email core action
- Submit for Approval core action

For example, if you use an email alert, manually add the email template that's used by that email alert. To deploy the package successfully, manually add those referenced components to the package.

For more information about actions, see the Actions Developer Guide.

Syntax

URI

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

Formats

JSON, XML

HTTP Methods

\ GET

Authentication

`Authorization: Bearer token`

Request parameters

None required

Example

Example Request

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/actions/custom -H
"Authorization: Bearer token"
```

Example Response Body

```
{
         "quickAction" : "/services/data/v60.0/actions/custom/quickAction",
        "apex" : "/services/data/v60.0/actions/custom/apex",
        "emailAlert" : "/services/data/v60.0/actions/custom/emailAlert",
```

```
        "flow" : "/services/data/v60.0/actions/custom/flow",
        "sendNotification" : "/services/data/v60.0/actions/custom/sendNotification"
         }
```

Return HTTP Headers for Custom Invocable Actions

Returns only the headers that are returned by sending a GET request to the custom invocable actions resource. This gives you a chance to see returned header values of the GET request before retrieving the content. This resource is available in REST API version 32.0 and later.

URI

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

Formats

JSON, XML

HTTP Methods

HEAD

Authentication

`Authorization: Bearer token`

Request parameters

None required

Example

Example Request

```
curl -X HEAD --head
https://MyDomainName.my.salesforce.com/services/data/v60.0/actions/custom -H "Authorization:
 Bearer token"
```

Example Response Body

```
HTTP/1.1 200 OK
Date: Mon, 21 Nov 2022 22:56:26 GMT
```

Invocable Actions Standard

Represents standard invocable actions that can be statically invoked. You can also get basic information for each type of action.

IN THIS SECTION:

Get Standard Invocable Actions

Gets the list of standard invocable actions that are provided by Salesforce. Some actions require special access. This resource is available in REST API version 32.0 and later.

Return HTTP Headers for Standard Invocable Actions

Returns only the headers that are returned by sending a GET request to the standard invocable actions resource. This gives you a chance to see returned header values of the GET request before retrieving the content. This resource is available in REST API version 32.0 and later.

SEE ALSO:

Apex Developer Guide : InvocableMethod Annotation

Get Standard Invocable Actions

Gets the list of standard invocable actions that are provided by Salesforce. Some actions require special access. This resource is available in REST API version 32.0 and later.

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

The Post to Chatter action supports the following features using a special format in the body post. For example, the string `Hi @[005000000000001], check out #[some_topic]` is stored appropriately as `Hi @Joe, check out #some_topic` where "@Joe" and "#some_topic" are links to the user and topic, respectively.

- @mentions using `@[<id>]`
- Topic links using `#[<topicString>]`

For more information about actions, see the *Actions Developer Guide*.

Syntax

URI

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

Formats

JSON, XML

HTTP Methods

GET

Authentication

`Authorization: Bearer token`

Request parameters

None required

Example

Example Request

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/actions/standard -H
"Authorization: Bearer token"
```

Example Response Body


```
{
  "actions" : [ {
    "label" : "Post to Chatter",
    "name" : "chatterPost",
    "type" : "CHATTERPOST",
    "url" : "/services/data/v60.0/actions/standard/chatterPost"
  }, {
    "label" : "Enable Folder Support for a Content Workspace (Library)",
    "name" : "contentWorkspaceEnableFolders",
    "type" : "CONTENTWORKSPACE_ENABLE_FOLDERS",
    "url" : "/services/data/v60.0/actions/standard/contentWorkspaceEnableFolders"
  }, {
    "label" : "Send Email",
    "name" : "emailSimple",
    "type" : "EMAILSIMPLE",
    "url" : "/services/data/v60.0/actions/standard/emailSimple"
  }, {
    "label" : "Submit for Approval",
    "name" : "submit",
    "type" : "SUBMITAPPROVAL",
    "url" : "/services/data/v60.0/actions/standard/submit"
  }, {
    "label" : "Deactivate Session-Based Permission Set",
    "name" : "deactivateSessionPermSet",
    "type" : "DEACTIVATE_SESSION_PERM_SET",
    "url" : "/services/data/v60.0/actions/standard/deactivateSessionPermSet"
  }, {
    "label" : "Activate Session-Based Permission Set",
    "name" : "activateSessionPermSet",
    "type" : "ACTIVATE_SESSION_PERM_SET",
    "url" : "/services/data/v60.0/actions/standard/activateSessionPermSet"
  }, {
    "label" : "Choose Price Book",
    "name" : "choosePricebook",
    "type" : "CHOOSE_PRICEBOOK",
    "url" : "/services/data/v60.0/actions/standard/choosePricebook"
  }, {
    "label" : "Routing Address Verification",
    "name" : "routingAddressVerification",
    "type" : "ROUTING_ADDRESS_VERIFICATION",
    "url" : "/services/data/v60.0/actions/standard/routingAddressVerification"
  }, {
```

```
    "label" : "Create Customer Contact Request",
    "name" : "contactRequestAction",
    "type" : "CONTACT_REQUEST_ACTION",
    "url" : "/services/data/v60.0/actions/standard/contactRequestAction"
  }, {
    "label" : "Publish Managed Content Release",
    "name" : "managedContentReleasePublish",
    "type" : "MANAGED_CONTENT_RELEASE_PUBLISH",
    "url" : "/services/data/v60.0/actions/standard/managedContentReleasePublish"
  } ]
}
```

Return HTTP Headers for Standard Invocable Actions

Returns only the headers that are returned by sending a GET request to the standard invocable actions resource. This gives you a chance to see returned header values of the GET request before retrieving the content. This resource is available in REST API version 32.0 and later.

Syntax

URI

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

Formats

JSON, XML

HTTP Methods

HEAD

Authentication

`Authorization: Bearer token`

Request parameters

None required

Example

Example Request

```
curl -X HEAD --head
https://MyDomainName.my.salesforce.com/services/data/v60.0/actions/standard -H
"Authorization: Bearer token"
```

Example Response Body

```
HTTP/1.1 200 OK
Date: Mon, 21 Nov 2022 22:56:26 GMT
```

List View Basic Information

Returns basic information for a specific list view, including the label, API name, and ID. This resource is available in REST API version 32.0 and later.

URI

`/services/data/vXX.X/sobjects/sObject/listviews/listViewID`

Formats

JSON, XML

HTTP Methods

GET

Authentication

`Authorization: Bearer token`

Parameters

None

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcBeMAK
 -H "Authorization: Bearer token"
```

Example Response Body

```
{
  "describeUrl" :
"/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcBeMAK/describe",
  "developerName" : "NewThisWeek",
  "id" : "00BD0000005WcBeMAK",
  "label" : "New This Week",
  "resultsUrl" :
"/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcBeMAK/results",
  "soqlCompatible" : true,
  "url" : "/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcBeMAK"
}
```

List View Describe

Returns detailed information about a list view, including the ID, the columns, and the SOQL query.

This resource is available in REST API version 32.0 and later.

URI

/services/data/v***XX.X***/sobjects/***sObject***/listviews/***queryLocator***/describe

Formats

JSON, XML

HTTP Method

GET

Authentication

Authorization: Bearer ***token***

Parameters

None

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcBeMAK/describe
 -H "Authorization: Bearer token"
```

Example Response Body


```
{
  "columns" : [ {
    "ascendingLabel" : "Z-A",
    "descendingLabel" : "A-Z",
    "fieldNameOrPath" : "Name",
    "hidden" : false,
    "label" : "Account Name",
    "selectListItem" : "Name",
    "sortDirection" : "ascending",
    "sortIndex" : 0,
    "sortable" : true,
    "type" : "string"
  }, {
    "ascendingLabel" : "Z-A",
    "descendingLabel" : "A-Z",
    "fieldNameOrPath" : "Site",
    "hidden" : false,
    "label" : "Account Site",
    "selectListItem" : "Site",
    "sortDirection" : null,
    "sortIndex" : null,
    "sortable" : true,
    "type" : "string"
  }, {
    "ascendingLabel" : "Z-A",
    "descendingLabel" : "A-Z",
    "fieldNameOrPath" : "BillingState",
    "hidden" : false,
    "label" : "Billing State/Province",
    "selectListItem" : "BillingState",
```

```
    "sortDirection" : null,
    "sortIndex" : null,
    "sortable" : true,
    "type" : "string"
  }, {
    "ascendingLabel" : "9-0",
    "descendingLabel" : "0-9",
    "fieldNameOrPath" : "Phone",
    "hidden" : false,
    "label" : "Phone",
    "selectListItem" : "Phone",
    "sortDirection" : null,
    "sortIndex" : null,
    "sortable" : true,
    "type" : "phone"
  }, {
    "ascendingLabel" : "Low to High",
    "descendingLabel" : "High to Low",
    "fieldNameOrPath" : "Type",
    "hidden" : false,
    "label" : "Type",
    "selectListItem" : "toLabel(Type)",
    "sortDirection" : null,
    "sortIndex" : null,
    "sortable" : true,
    "type" : "picklist"
  }, {
    "ascendingLabel" : "Z-A",
    "descendingLabel" : "A-Z",
    "fieldNameOrPath" : "Owner.Alias",
    "hidden" : false,
    "label" : "Account Owner Alias",
    "selectListItem" : "Owner.Alias",
    "sortDirection" : null,
    "sortIndex" : null,
    "sortable" : true,
    "type" : "string"
  }, {
    "ascendingLabel" : null,
    "descendingLabel" : null,
    "fieldNameOrPath" : "Id",
    "hidden" : true,
    "label" : "Account ID",
    "selectListItem" : "Id",
    "sortDirection" : null,
    "sortIndex" : null,
    "sortable" : false,
    "type" : "id"
  }, {
    "ascendingLabel" : null,
    "descendingLabel" : null,
    "fieldNameOrPath" : "CreatedDate",
    "hidden" : true,
    "label" : "Created Date",
```

```
    "selectListItem" : "CreatedDate",
    "sortDirection" : null,
    "sortIndex" : null,
    "sortable" : false,
    "type" : "datetime"
  }, {
    "ascendingLabel" : null,
    "descendingLabel" : null,
    "fieldNameOrPath" : "LastModifiedDate",
    "hidden" : true,
    "label" : "Last Modified Date",
    "selectListItem" : "LastModifiedDate",
    "sortDirection" : null,
    "sortIndex" : null,
    "sortable" : false,
    "type" : "datetime"
  }, {
    "ascendingLabel" : null,
    "descendingLabel" : null,
    "fieldNameOrPath" : "SystemModstamp",
    "hidden" : true,
    "label" : "System Modstamp",
    "selectListItem" : "SystemModstamp",
    "sortDirection" : null,
    "sortIndex" : null,
    "sortable" : false,
    "type" : "datetime"
  } ],
  "id" : "00BD0000005WcBe",
  "orderBy" : [ {
    "fieldNameOrPath" : "Name",
    "nullsPosition" : "first",
    "sortDirection" : "ascending"
  }, {
    "fieldNameOrPath" : "Id",
    "nullsPosition" : "first",
    "sortDirection" : "ascending"
  } ],
  "query" : "SELECT name, site, billingstate, phone, tolabel(type), owner.alias, id,
createddate, lastmodifieddate, systemmodstamp FROM Account WHERE CreatedDate = THIS_WEEK
 ORDER BY Name ASC NULLS FIRST, Id ASC NULLS FIRST",
  "scope" : null,
  "sobjectType" : "Account",
  "whereCondition" : {
    "field" : "CreatedDate",
    "operator" : "equals",
    "values" : [ "THIS_WEEK" ]
  }
}
```

List View Results

Executes the SOQL query for the list view and returns the resulting data and presentation information. This resource is available in REST API version 32.0 and later.

Syntax

URI

`/services/data/vXX.X/sobjects/sObject/listviews/listViewID/results`

Formats

JSON, XML

HTTP Method

GET

Authentication

`Authorization: Bearer token`

Parameters

Parameter	Description
`limit`	The maximum number of records to return, between 1-2000. The default value is 25.
`offset`	The first record to return. Use this parameter to paginate the results. The default value is 1.

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcCNMA0/results
 -H "Authorization: Bearer token"
```

Example Response Body


```
{
  "columns" : [ {
    "ascendingLabel" : "Z-A",
    "descendingLabel" : "A-Z",
    "fieldNameOrPath" : "Name",
    "hidden" : false,
    "label" : "Account Name",
    "selectListItem" : "Name",
    "sortDirection" : "ascending",
    "sortIndex" : 0,
    "sortable" : true,
    "type" : "string"
  }, {
    "ascendingLabel" : "Z-A",
```

```
  "descendingLabel" : "A-Z",
  "fieldNameOrPath" : "Site",
  "hidden" : false,
  "label" : "Account Site",
  "selectListItem" : "Site",
  "sortDirection" : null,
  "sortIndex" : null,
  "sortable" : true,
  "type" : "string"
}, {
  "ascendingLabel" : "Z-A",
  "descendingLabel" : "A-Z",
  "fieldNameOrPath" : "BillingState",
  "hidden" : false,
  "label" : "Billing State/Province",
  "selectListItem" : "BillingState",
  "sortDirection" : null,
  "sortIndex" : null,
  "sortable" : true,
  "type" : "string"
}, {
  "ascendingLabel" : "9-0",
  "descendingLabel" : "0-9",
  "fieldNameOrPath" : "Phone",
  "hidden" : false,
  "label" : "Phone",
  "selectListItem" : "Phone",
  "sortDirection" : null,
  "sortIndex" : null,
  "sortable" : true,
  "type" : "phone"
}, {
  "ascendingLabel" : "Low to High",
  "descendingLabel" : "High to Low",
  "fieldNameOrPath" : "Type",
  "hidden" : false,
  "label" : "Type",
  "selectListItem" : "toLabel(Type)",
  "sortDirection" : null,
  "sortIndex" : null,
  "sortable" : true,
  "type" : "picklist"
}, {
  "ascendingLabel" : "Z-A",
  "descendingLabel" : "A-Z",
  "fieldNameOrPath" : "Owner.Alias",
  "hidden" : false,
  "label" : "Account Owner Alias",
  "selectListItem" : "Owner.Alias",
  "sortDirection" : null,
  "sortIndex" : null,
  "sortable" : true,
  "type" : "string"
}, {
```

```
  "ascendingLabel" : null,
  "descendingLabel" : null,
  "fieldNameOrPath" : "Id",
  "hidden" : true,
  "label" : "Account ID",
  "selectListItem" : "Id",
  "sortDirection" : null,
  "sortIndex" : null,
  "sortable" : false,
  "type" : "id"
}, {
  "ascendingLabel" : null,
  "descendingLabel" : null,
  "fieldNameOrPath" : "CreatedDate",
  "hidden" : true,
  "label" : "Created Date",
  "selectListItem" : "CreatedDate",
  "sortDirection" : null,
  "sortIndex" : null,
  "sortable" : false,
  "type" : "datetime"
}, {
  "ascendingLabel" : null,
  "descendingLabel" : null,
  "fieldNameOrPath" : "LastModifiedDate",
  "hidden" : true,
  "label" : "Last Modified Date",
  "selectListItem" : "LastModifiedDate",
  "sortDirection" : null,
  "sortIndex" : null,
  "sortable" : false,
  "type" : "datetime"
}, {
  "ascendingLabel" : null,
  "descendingLabel" : null,
  "fieldNameOrPath" : "SystemModstamp",
  "hidden" : true,
  "label" : "System Modstamp",
  "selectListItem" : "SystemModstamp",
  "sortDirection" : null,
  "sortIndex" : null,
  "sortable" : false,
  "type" : "datetime"
} ],
"developerName" : "MyAccounts",
"done" : true,
"id" : "00BD0000005WcCN",
"label" : "My Accounts",
"records" : [ {
  "columns" : [ {
    "fieldNameOrPath" : "Name",
    "value" : "Burlington Textiles Corp of America"
  }, {
    "fieldNameOrPath" : "Site",
```

```
      "value" : null
    }, {
      "fieldNameOrPath" : "BillingState",
      "value" : "NC"
    }, {
      "fieldNameOrPath" : "Phone",
      "value" : "(336) 222-7000"
    }, {
      "fieldNameOrPath" : "Type",
      "value" : "Customer - Direct"
    }, {
      "fieldNameOrPath" : "Owner.Alias",
      "value" : "TUser"
    }, {
      "fieldNameOrPath" : "Id",
      "value" : "001D000000JliSTIAZ"
    }, {
      "fieldNameOrPath" : "CreatedDate",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    }, {
      "fieldNameOrPath" : "LastModifiedDate",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    }, {
      "fieldNameOrPath" : "SystemModstamp",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    } ]
  }, {
    "columns" : [ {
      "fieldNameOrPath" : "Name",
      "value" : "Dickenson plc"
    }, {
      "fieldNameOrPath" : "Site",
      "value" : null
    }, {
      "fieldNameOrPath" : "BillingState",
      "value" : "KS"
    }, {
      "fieldNameOrPath" : "Phone",
      "value" : "(785) 241-6200"
    }, {
      "fieldNameOrPath" : "Type",
      "value" : "Customer - Channel"
    }, {
      "fieldNameOrPath" : "Owner.Alias",
      "value" : "TUser"
    }, {
      "fieldNameOrPath" : "Id",
      "value" : "001D000000JliSVIAZ"
    }, {
      "fieldNameOrPath" : "CreatedDate",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    }, {
      "fieldNameOrPath" : "LastModifiedDate",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
```

```
    }, {
      "fieldNameOrPath" : "SystemModstamp",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    } ]
  }, {
    "columns" : [ {
      "fieldNameOrPath" : "Name",
      "value" : "Edge Communications"
    }, {
      "fieldNameOrPath" : "Site",
      "value" : null
    }, {
      "fieldNameOrPath" : "BillingState",
      "value" : "TX"
    }, {
      "fieldNameOrPath" : "Phone",
      "value" : "(512) 757-6000"
    }, {
      "fieldNameOrPath" : "Type",
      "value" : "Customer - Direct"
    }, {
      "fieldNameOrPath" : "Owner.Alias",
      "value" : "TUser"
    }, {
      "fieldNameOrPath" : "Id",
      "value" : "001D000000JliSSIAZ"
    }, {
      "fieldNameOrPath" : "CreatedDate",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    }, {
      "fieldNameOrPath" : "LastModifiedDate",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    }, {
      "fieldNameOrPath" : "SystemModstamp",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    } ]
  }, {
    "columns" : [ {
      "fieldNameOrPath" : "Name",
      "value" : "Express Logistics and Transport"
    }, {
      "fieldNameOrPath" : "Site",
      "value" : null
    }, {
      "fieldNameOrPath" : "BillingState",
      "value" : "OR"
    }, {
      "fieldNameOrPath" : "Phone",
      "value" : "(503) 421-7800"
    }, {
      "fieldNameOrPath" : "Type",
      "value" : "Customer - Channel"
    }, {
      "fieldNameOrPath" : "Owner.Alias",
```

```
      "value" : "TUser"
    }, {
      "fieldNameOrPath" : "Id",
      "value" : "001D000000JliSXIAZ"
    }, {
      "fieldNameOrPath" : "CreatedDate",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    }, {
      "fieldNameOrPath" : "LastModifiedDate",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    }, {
      "fieldNameOrPath" : "SystemModstamp",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    } ]
  }, {
    "columns" : [ {
      "fieldNameOrPath" : "Name",
      "value" : "GenePoint"
    }, {
      "fieldNameOrPath" : "Site",
      "value" : null
    }, {
      "fieldNameOrPath" : "BillingState",
      "value" : "CA"
    }, {
      "fieldNameOrPath" : "Phone",
      "value" : "(650) 867-3450"
    }, {
      "fieldNameOrPath" : "Type",
      "value" : "Customer - Channel"
    }, {
      "fieldNameOrPath" : "Owner.Alias",
      "value" : "TUser"
    }, {
      "fieldNameOrPath" : "Id",
      "value" : "001D000000JliSPIAZ"
    }, {
      "fieldNameOrPath" : "CreatedDate",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    }, {
      "fieldNameOrPath" : "LastModifiedDate",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    }, {
      "fieldNameOrPath" : "SystemModstamp",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    } ]
  }, {
    "columns" : [ {
      "fieldNameOrPath" : "Name",
      "value" : "Grand Hotels and Resorts Ltd"
    }, {
      "fieldNameOrPath" : "Site",
      "value" : null
    }, {
```

```
      "fieldNameOrPath" : "BillingState",
      "value" : "IL"
    }, {
      "fieldNameOrPath" : "Phone",
      "value" : "(312) 596-1000"
    }, {
      "fieldNameOrPath" : "Type",
      "value" : "Customer - Direct"
    }, {
      "fieldNameOrPath" : "Owner.Alias",
      "value" : "TUser"
    }, {
      "fieldNameOrPath" : "Id",
      "value" : "001D000000JliSWIAZ"
    }, {
      "fieldNameOrPath" : "CreatedDate",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    }, {
      "fieldNameOrPath" : "LastModifiedDate",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    }, {
      "fieldNameOrPath" : "SystemModstamp",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    } ]
  }, {
    "columns" : [ {
      "fieldNameOrPath" : "Name",
      "value" : "Pyramid Construction Inc."
    }, {
      "fieldNameOrPath" : "Site",
      "value" : null
    }, {
      "fieldNameOrPath" : "BillingState",
      "value" : null
    }, {
      "fieldNameOrPath" : "Phone",
      "value" : "(014) 427-4427"
    }, {
      "fieldNameOrPath" : "Type",
      "value" : "Customer - Channel"
    }, {
      "fieldNameOrPath" : "Owner.Alias",
      "value" : "TUser"
    }, {
      "fieldNameOrPath" : "Id",
      "value" : "001D000000JliSUIAZ"
    }, {
      "fieldNameOrPath" : "CreatedDate",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    }, {
      "fieldNameOrPath" : "LastModifiedDate",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    }, {
      "fieldNameOrPath" : "SystemModstamp",
```

```
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    } ]
  }, {
    "columns" : [ {
      "fieldNameOrPath" : "Name",
      "value" : "sForce"
    }, {
      "fieldNameOrPath" : "Site",
      "value" : null
    }, {
      "fieldNameOrPath" : "BillingState",
      "value" : "CA"
    }, {
      "fieldNameOrPath" : "Phone",
      "value" : "(415) 901-7000"
    }, {
      "fieldNameOrPath" : "Type",
      "value" : null
    }, {
      "fieldNameOrPath" : "Owner.Alias",
      "value" : "TUser"
    }, {
      "fieldNameOrPath" : "Id",
      "value" : "001D000000JliSaIAJ"
    }, {
      "fieldNameOrPath" : "CreatedDate",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    }, {
      "fieldNameOrPath" : "LastModifiedDate",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    }, {
      "fieldNameOrPath" : "SystemModstamp",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    } ]
  }, {
    "columns" : [ {
      "fieldNameOrPath" : "Name",
      "value" : "United Oil and Gas Corp."
    }, {
      "fieldNameOrPath" : "Site",
      "value" : null
    }, {
      "fieldNameOrPath" : "BillingState",
      "value" : "NY"
    }, {
      "fieldNameOrPath" : "Phone",
      "value" : "(212) 842-5500"
    }, {
      "fieldNameOrPath" : "Type",
      "value" : "Customer - Direct"
    }, {
      "fieldNameOrPath" : "Owner.Alias",
      "value" : "TUser"
    }, {
```

```
      "fieldNameOrPath" : "Id",
      "value" : "001D000000JliSZIAZ"
    }, {
      "fieldNameOrPath" : "CreatedDate",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    }, {
      "fieldNameOrPath" : "LastModifiedDate",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    }, {
      "fieldNameOrPath" : "SystemModstamp",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    } ]
  }, {
    "columns" : [ {
      "fieldNameOrPath" : "Name",
      "value" : "United Oil and Gas, Singapore"
    }, {
      "fieldNameOrPath" : "Site",
      "value" : null
    }, {
      "fieldNameOrPath" : "BillingState",
      "value" : "Singapore"
    }, {
      "fieldNameOrPath" : "Phone",
      "value" : "(650) 450-8810"
    }, {
      "fieldNameOrPath" : "Type",
      "value" : "Customer - Direct"
    }, {
      "fieldNameOrPath" : "Owner.Alias",
      "value" : "TUser"
    }, {
      "fieldNameOrPath" : "Id",
      "value" : "001D000000JliSRIAZ"
    }, {
      "fieldNameOrPath" : "CreatedDate",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    }, {
      "fieldNameOrPath" : "LastModifiedDate",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    }, {
      "fieldNameOrPath" : "SystemModstamp",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    } ]
  }, {
    "columns" : [ {
      "fieldNameOrPath" : "Name",
      "value" : "United Oil and Gas, UK"
    }, {
      "fieldNameOrPath" : "Site",
      "value" : null
    }, {
      "fieldNameOrPath" : "BillingState",
      "value" : "UK"
```

```
    }, {
      "fieldNameOrPath" : "Phone",
      "value" : "+44 191 4956203"
    }, {
      "fieldNameOrPath" : "Type",
      "value" : "Customer - Direct"
    }, {
      "fieldNameOrPath" : "Owner.Alias",
      "value" : "TUser"
    }, {
      "fieldNameOrPath" : "Id",
      "value" : "001D000000JliSQIAZ"
    }, {
      "fieldNameOrPath" : "CreatedDate",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    }, {
      "fieldNameOrPath" : "LastModifiedDate",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    }, {
      "fieldNameOrPath" : "SystemModstamp",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    } ]
  }, {
    "columns" : [ {
      "fieldNameOrPath" : "Name",
      "value" : "University of Arizona"
    }, {
      "fieldNameOrPath" : "Site",
      "value" : null
    }, {
      "fieldNameOrPath" : "BillingState",
      "value" : "AZ"
    }, {
      "fieldNameOrPath" : "Phone",
      "value" : "(520) 773-9050"
    }, {
      "fieldNameOrPath" : "Type",
      "value" : "Customer - Direct"
    }, {
      "fieldNameOrPath" : "Owner.Alias",
      "value" : "TUser"
    }, {
      "fieldNameOrPath" : "Id",
      "value" : "001D000000JliSYIAZ"
    }, {
      "fieldNameOrPath" : "CreatedDate",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    }, {
      "fieldNameOrPath" : "LastModifiedDate",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    }, {
      "fieldNameOrPath" : "SystemModstamp",
      "value" : "Fri Aug 01 21:15:46 GMT 2014"
    } ]
```

```
  } ],
  "size" : 12
}
```

List Views for an Object

Returns the list of list views for the specified sObject, including the ID and other basic information about each list view. This resource is available in REST API version 32.0 and later.

URI

`/services/data/vXX.X/sobjects/sObject/listviews`

Formats

JSON, XML

HTTP Methods

GET

Authentication

`Authorization: Bearer token`

Parameters

None

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/listviews
-H "Authorization: Bearer token"
```

Example Response Body


```
{
  "done" : true,
  "listviews" : [ {
    "describeUrl" :
"/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcBeMAK/describe",
    "developerName" : "NewThisWeek",
    "id" : "00BD0000005WcBeMAK",
    "label" : "New This Week",
    "resultsUrl" :
"/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcBeMAK/results",
    "soqlCompatible" : true,
    "url" : "/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcBeMAK"
  }, {
    "describeUrl" :
"/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcBpMAK/describe",
    "developerName" : "NewLastWeek",
    "id" : "00BD0000005WcBpMAK",
    "label" : "New Last Week",
    "resultsUrl" :
"/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcBpMAK/results",
```

```
    "soqlCompatible" : true,
    "url" : "/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcBpMAK"
  }, {
    "describeUrl" :
"/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcC6MAK/describe",
    "developerName" : "PlatinumandGoldSLACustomers",
    "id" : "00BD0000005WcC6MAK",
    "label" : "Platinum and Gold SLA Customers",
    "resultsUrl" :
"/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcC6MAK/results",
    "soqlCompatible" : true,
    "url" : "/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcC6MAK"
  }, {
    "describeUrl" :
"/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcCEMA0/describe",
    "developerName" : "RecentlyViewedAccounts",
    "id" : "00BD0000005WcCEMA0",
    "label" : "Recently Viewed Accounts",
    "resultsUrl" :
"/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcCEMA0/results",
    "soqlCompatible" : true,
    "url" : "/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcCEMA0"
  }, {
    "describeUrl" :
"/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcCFMA0/describe",
    "developerName" : "AllAccounts",
    "id" : "00BD0000005WcCFMA0",
    "label" : "All Accounts",
    "resultsUrl" :
"/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcCFMA0/results",
    "soqlCompatible" : true,
    "url" : "/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcCFMA0"
  }, {
    "describeUrl" :
"/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcCNMA0/describe",
    "developerName" : "MyAccounts",
    "id" : "00BD0000005WcCNMA0",
    "label" : "My Accounts",
    "resultsUrl" :
"/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcCNMA0/results",
    "soqlCompatible" : true,
    "url" : "/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcCNMA0"
  } ],
  "nextRecordsUrl" : null,
  "size" : 6,
  "sobjectType" : "Account"
}
```


Support Knowledge with REST API

Knowledge Support REST APIs allow both authorized and guest users to retrieve the user's visible data categories and their associated articles. This resource is available in REST API version 38.0 and later.

Authenticated users need the `UserProfile.apiEnabled` permission, Knowledge enabled in the organization, read rights on the article type, and any other knowledge specific permission or preference that controls visibility to articles.

Guest users need the `Guest Access to the Support API` preference enabled on the relevant Site, Knowledge enabled in the organization, and read rights on the article type and article channel that controls the visibility for guest users.

Syntax

URI

`/services/data/vXX.X/support`

Method

GET

Formats

JSON, XML

Authentication

`Authorization: Bearer token`

Example

Example Response Body

```
{
"dataCategoryGroups" : "/services/data/vXX.X/support/dataCategoryGroups",
"knowledgeArticles" : "/services/data/vXX.X/support/knowledgeArticles"
:
}
```

IN THIS SECTION:

Data Category Groups

Get data category groups that are visible to the current user. This resource is available in REST API version 38.0 and later.

Data Category Detail

Gets data category details and the child categories by a given category. This resource can be used in API version 38.0 and later.

Articles List

Get a page of online articles for the given language and category through either search or query. This resource is available in REST API version 38.0 and later.

Articles Details

Gets all online article fields, accessible to the user. This resource is available with article IDs in REST API version 38.0 and later, and this resource is available with article URL names in version 44.0 and later.

Data Category Groups

Get data category groups that are visible to the current user. This resource is available in REST API version 38.0 and later.

Salesforce Knowledge must be enabled in your organization. This resource can be used in API version 38.0 and later. Use the language code format used in *Which Languages Does Salesforce Support?*.

Only the user's visible data categories are returned. A user might be able to see several sub trees in the category group, therefore, the top categories that are visible to the user in each group are returned.

Syntax

URI

`/services/data/vXX.X/support/dataCategoryGroups`

Method

GET

Formats

JSON, XML

Authentication

`Authorization: Bearer token`

HTTP headers

Accept: Optional. Can be either `application/json` or `application/xml`.

Accept-language: Optional. Language to translate the categories. Any ISO-639 language abbreviation, and an ISO-3166 country code subtag in the HTTP Accept-Language header. Only one language accepted. If no language specified, the non-translated labels are returned.

Input:

string `sObjectName`: Required. `KnowledgeArticleVersion` only.

boolean `topCategoriesOnly`: Optional. Defaults to true

- True returns only the top level categories.
- False returns the entire tree.

Note: All the input parameters are case-sensitive.

Output:

A list of the active data category groups that are visible to the current user in the site context. Returns id, name, label, and their top level categories or the entire data category group tree that are visible to the current user. The labels must be translated to the given language if they are available.

- **Data Category Group List**

  This payload lists the active root Data Category Groups that can be used in other requests to return the data categories and articles related to it.

```
{
    "categoryGroups": [ Data Category Group, ....],
}
```

  Note: Returns only the active groups that are associated to the given entity (by `sObjectName`). Only `KnowledgeArticleVersion` is supported.

- **Data Category Group**

  This represents an individual data category group, and its root category.

```
{
    "name": String,  // the unique name of the category group
    "label": String,  // returns the translated version if it is available
```

```
    "objectUsage" : String, // currently only "KnowledgeArticleVersion" is available.

     "topCategories": [ Data Category Summary, ....]
}
```

- **Data Category Summary**

  This provides a summary of data category information. The Summary and Detail responses share common properties, with the goal of providing only as much information as is necessary from associated resources.

```
{
    "name": String, // the unique name of the category
    "label":  String,  // returns the translated version if it is available
    "url":  URL,    // the url points to the data category detail API
 "childCategories": [ Data Category Summary, ....] // null if topCategoriesOnly is
true
}
```

Note: The URL property is a pre-calculated path to the unique resource representing this data category, in this case it is a Data Category Detail API.

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/support/dataCategoryGroups?sObjectName=KnowledgeArticleVersion
 -H "Authorization: Bearer token"
```

Example Response Body

```
{
  "categoryGroups" : [ {
 "label" : "Doc",
 "name" : "Doc",
 "objectUsage" : "KnowledgeArticleVersion",
 "topCategories" : [ {
   "childCategories" : null,
   "label" : "All",
   "name" : "All",
   "url" :
"/services/data/v60.0/support/dataCategoryGroups/Doc/dataCategories/All?sObjectName=KnowledgeArticleVersion"

 } ]
  }, {
 "label" : "Manual",
 "name" : "Manual",
 "objectUsage" : "KnowledgeArticleVersion",
 "topCategories" : [ {
   "childCategories" : null,
   "label" : "All",
   "name" : "All",
   "url" :
"/services/data/v60.0/support/dataCategoryGroups/Manual/dataCategories/All?sObjectName=KnowledgeArticleVersion"
```

```
    } ]
  } ]
}
```

Data Category Detail

Gets data category details and the child categories by a given category. This resource can be used in API version 38.0 and later.

Salesforce Knowledge must be enabled in your organization. Use the language code format used in *Which Languages Does Salesforce Support?*.

Syntax

URI

`/services/data/vXX.X/support/dataCategoryGroups/group/dataCategories/category`

Method

GET

Formats

JSON, XML

Authentication

`Authorization: Bearer token`

HTTP headers

Accept: Optional. Can be either `application/json` or `application/xml`.

Accept-language: Optional. Language to translate the categories. Any ISO-639 language abbreviation, and an ISO-3166 country code subtag in the HTTP Accept-Language header. Only one language accepted. If no language specified, the non-translated labels are returned.

Input:

string `sObjectName`: Required. `KnowledgeArticleVersion` only.

Output:

Details of the category and a list of child categories (name, label, etc.).

- **Data Category Detail**

  Used for situations where the hierarchical representation of data categories is important. The child property contains a list of child data categories.

```
{
    "name": String, // the unique name of the category
    "label":  String, // returns the translated version if it is available
    "url":  URL,
    "childCategories": [ Data Category Summary, ....],
}
```

Note: If the category isn't visible to the current user the return is empty.

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/support/dataCategoryGroups/Doc/dataCategories/All?sObjectName=KnowledgeArticleVersion
 -H "Authorization: Bearer token"
```

Example Response Body

```
{
  "childCategories" : [ {
    "childCategories" : null,
    "label" : "Help",
    "name" : "Help",
    "url" :
"/services/data/v60.0/support/dataCategoryGroups/Doc/dataCategories/Help?sObjectName=KnowledgeArticleVersion"

  }, {
    "childCategories" : null,
    "label" : "QA",
    "name" : "QA",
    "url" :
"/services/data/v60.0/support/dataCategoryGroups/Doc/dataCategories/QA?sObjectName=KnowledgeArticleVersion"

  } ],
  "label" : "All",
  "name" : "All",
  "url" :
"/services/data/v60.0/support/dataCategoryGroups/Doc/dataCategories/All?sObjectName=KnowledgeArticleVersion"
}
```

Articles List

Get a page of online articles for the given language and category through either search or query. This resource is available in REST API version 38.0 and later.

Syntax

URI

`/services/data/vXX.X/support/knowledgeArticles`

Method

GET

Formats

JSON, XML

Authentication

`Authorization: Bearer token`

HTTP headers

Accept: Optional. Can be either `application/json` or `application/xml`.

Accept-language: Required. The article must be an active language in the user’s organization

- If the language code isn't valid, an error message is returned: "The language code is not valid or not supported by Knowledge."
- If the language code is valid, but not supported by Knowledge, then an error message is returned: "Invalid language code. Check that the language is included in your Knowledge language settings."

Input:

string `q`: Optional, Performs an SOSL search. If the query string is null, empty, or not given, an SOQL query runs.

The characters `?` and `*` are used for wildcard searches. The characters `(`, `)`, and `"` are used for complex search terms. See https://developer.salesforce.com/docs/atlas.en-us.soql_sosl.meta/soql_sosl/sforce_api_calls_sosl_find.htm.

string `channel`: Optional, defaults to user's context. For information on channel values, see Valid `channel` values.

- **App**: Visible in the internal Salesforce Knowledge application
- **Pkb**: Visible in the public knowledge base
- **Csp**: Visible in the Customer Portal
- **Prm**: Visible in the Partner Portal

string `categories` in map json format `{"group1":"category1","group2":"category2",...} )`

Optional, defaults to None. Category group must be unique in each group:category pair, otherwise you get `ARGUMENT_OBJECT_PARSE_ERROR`. There is a limit of three data category conditions, otherwise you get `INVALID_FILTER_VALUE`.

string `queryMethod` values are: `AT, BELOW, ABOVE, ABOVE_OR_BELOW`. Only valid when categories are specified, defaults to ABOVE_OR_BELOW.

string `sort`: Optional, a sortable field name `LastPublishedDate, CreatedDate, Title, ViewScore`. Defaults to `LastPublishedDate` for query and relevance for search.

Note: When sorting on `ViewScore` it is only available for query, not search, and no pagination is supported. You only get one page of results.

string `order`: Optional, either ASC or DESC, defaults to DESC. Valid only when sort is valid.

integer `pageSize`: Optional, defaults to 20. Valid range 1 to 100.

integer `pageNumber` : Optional, defaults to 1.

Output:

A page of online articles in the given language and category visible to the current user.

- **Article Page**

  A page of articles. The individual entries are article summaries so the size can be kept at a minimum.

```
{
   "articles": [ Article Summary, … ],  // list of articles
      "currentPageUrl": URL,   // the article list API with current page number
        "nextPageUrl": URL,  // the article list API with next page number,
      which can be null if there are no more articles.
      "pageNumber": Int  // the current page number, starting at 1.
    }
```

  Note: The API supports paging. Each page of responses includes a URL to its page, as well as the URL to the next page of articles.

  Note: if the user input parameter has the default value, the API does not show this parameter in either `currentPageUrl` or `nextPageUrl`.

- **Article Summary**

A summary of an article used in a list of article responses. It shares similar properties to the Article Detail representation, as one is a superset of the other.

```
{
      "id":  Id,    // articleId
      "articleNumber": String,
      "articleType": String,  // apiName of the article type, available in API v44.0
 and later
      "title":  String,
      "urlName": String,  // available in API v44.0 and later
      "summary":  String,
      "url":  URL,  // to the Article Detail API
      "viewCount": Int,       // view count in the interested channel
      "viewScore": double (in xxx.xxxx precision),   // view score in the interested
 channel.
      "upVoteCount": int,  // up vote count in the interested channel.
      "downVoteCount": int, // down vote count in the interested channel.
      "lastPublishedDate": Date // last publish date in ISO8601 format
      "categoryGroups":  [ Data Category Group, …. ]}
```

The "url" property always points to the Article Details resource endpoint. For information on valid channel values, see the channel parameter description

- **Data Category Group**

  An individual data category group, its root category, and a list of selected data categories in the group.

```
{
      "groupName": String,  // the unique name of the category group
      "groupLabel": String,  // returns the translated version
      "selectedCategories": [ Data Category Summary, … ]
    }
```

- **Data Category Summary**

  Provides a summary of data category information. The Summary and Detail responses share common properties.

```
{
            "categoryName": String, // the unique name of the category
            "categoryLabel":  String, // returns the translated version, per the API
 language specified
            "url":  String // returns the url for the DataCategory REST API.
    }
```

Note: The outputs of Data Category Group and Data Category Summary in Article List API are different from the Data Category Groups API.

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/support/knowledgeArticles?sort=ViewScore&channel=Pkb&pageSize=3

    HTTP Headers:
        Content-Type: application/json; charset=UTF-8
```

```
        Accept: application/json
        Accept-Language: en-US
```

Example Response Body


```
{
 "articles" : [ {
 "articleNumber" : "000001002",
 "categoryGroups" : [ ],
 "downVoteCount" : 0,
 "id" : "kA0xx000000000BCAQ",
 "lastPublishedDate" : "2015-02-25T02:07:18Z",
 "summary" : "With this online Chinese input tool, you can type Chinese characters
through your web browser without installing any Chinese input software in your system.
 The Chinese online input tool uses the popular Pin Yin input method. It is a fast and
 convenient tool to input Chinese on English OS environments.",
 "title" : "Long text test",
 "upVoteCount" : 0,
 "url" : "/services/data/v60.0/support/knowledgeArticles/kA0xx000000000BCAQ",
 "viewCount" : 4,
 "viewScore" : 100.0
 }, {
 "articleNumber" : "000001004",
 "categoryGroups" : [ ],
 "downVoteCount" : 0,
 "id" : "kA0xx000000000LCAQ",
 "lastPublishedDate" : "2016-06-21T21:11:02Z",
 "summary" : "The number of characters required for complete coverage of all these
languages' needs cannot fit in the 256-character code space of 8-bit character encodings,
 requiring at least a 16-bit fixed width encoding or multi-byte variable-length encodings.
 \r\n\r\nAlthough CJK encodings have common character sets, the encodings often used to
 represent them have been developed separately by different East Asian governments and
 software companies, and are mutually incompatible. Unicode has attempted, with some
controversy, to unify the character sets in a process known as Han unification.\r\n\r\nCJK
 character encodings should consist minimally of Han characters p",
 "title" : "Test Images",
 "upVoteCount" : 0,
 "url" : "/services/data/v60.0/support/knowledgeArticles/kA0xx000000000LCAQ",
 "viewCount" : 0,
 "viewScore" : 0.0
 }, {
 "articleNumber" : "000001012",
 "categoryGroups" : [ ],
 "downVoteCount" : 0,
 "id" : "kA0xx000000006GCAQ",
 "lastPublishedDate" : "2016-06-21T21:10:48Z",
 "summary" : null,
 "title" : "Test Draft 2",
 "upVoteCount" : 0,
 "url" : "/services/data/v60.0/support/knowledgeArticles/kA0xx000000006GCAQ",
 "viewCount" : 0,
 "viewScore" : 0.0
 } ],
 "currentPageUrl" :
"/services/data/v60.0/support/knowledgeArticles?channel=Pkb&amp;pageSize=3&amp;sort=ViewScore",
```

```
 "nextPageUrl" : null,
 "pageNumber" : 1
}
```

Usage

Salesforce Knowledge must be enabled in your organization. This resource can be used in API version 38.0 and later. The Custom File Field is not supported because it returns a link to a binary stream. Use the language code format used in *Which Languages Does Salesforce Support?*.

Valid `channel` Values

- When using the options string `channel`, where the matching articles are visible, the following values are valid.
  - `App`–Visible in the internal Salesforce Knowledge application
  - `Pkb`–Visible in the public knowledge base
  - `Csp`–Visible in the Customer Portal
  - `Prm`–Visible in the Partner Portal
- If `channel` isn't specified, the default value is determined by the type of user.
  - `Pkb` for a guest user
  - `Csp` for a Customer Portal user
  - `Prm` for a Partner Portal user
  - `App` for any other type of user
- If `channel` is specified, the specified value may be used to retrieve articles.
  - For guest, Customer Portal, and Partner Portal users, if the specified channel is other than the channel accessible to the user, an error is returned.
  - For all users other than guest, Customer Portal, and Partner Portal users, the specified channel value is used.

Articles Details

Gets all online article fields, accessible to the user. This resource is available with article IDs in REST API version 38.0 and later, and this resource is available with article URL names in version 44.0 and later.

Salesforce Knowledge must be enabled in your organization. This resource can be used in API version 38.0 and later. The Custom File Field is not supported because it returns a link to a binary stream. Use the language code format used in *Which Languages Does Salesforce Support?*.

A lookup custom field is visible to guest users depending on the lookup entity type. For example, User is visible, but Case and Account are not visible. Following standard fields are not visible to a guest user, even if they are in the layout:

- archivedBy
- isLatestVersion
- translationCompletedDate
- translationImportedDate
- translationExportedDate
- versionNumber

- visibleInInternalApp
- visibleInPKB
- visibleToCustomer
- visbileToPartner

Valid `channel` Values

- When using the options string `channel`, where the matching articles are visible, the following values are valid.
  - `App`–Visible in the internal Salesforce Knowledge application
  - `Pkb`–Visible in the public knowledge base
  - `Csp`–Visible in the Customer Portal
  - `Prm`–Visible in the Partner Portal
- If `channel` isn't specified, the default value is determined by the type of user.
  - `Pkb` for a guest user
  - `Csp` for a Customer Portal user
  - `Prm` for a Partner Portal user
  - `App` for any other type of user
- If `channel` is specified, the specified value may be used to retrieve articles.
  - For guest, Customer Portal, and Partner Portal users, if the specified channel is other than the channel accessible to the user, an error is returned.
  - For all users other than guest, Customer Portal, and Partner Portal users, the specified channel value is used.

Syntax

Method

GET

Formats

JSON, XML

Authentication

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

Endpoint

`/services/data/v`***`XX.X`***`/support/knowledgeArticles/`***`articleId_or_articleUrlName`***

HTTP headers

Accept: Optional. Can be either `application/json` or `application/xml`.

Accept-language: Required. The article must be an active language in the user's organization

- If the language code isn't valid, an error message is returned: "The language code is not valid or not supported by Knowledge."
- If the language code is valid, but not supported by Knowledge, then an error message is returned: "Invalid language code. Check that the language is included in your Knowledge language settings."

Input:

string `channel`: Optional, defaults to user's context. For information on channel values, see Valid `channel` Values.

- **App**: Visible in the internal Salesforce Knowledge application

- **Pkb**: Visible in the public knowledge base
- **Csp**: Visible in the Customer Portal
- **Prm**: Visible in the Partner Portal

boolean `updateViewStat`: Optional, defaults to true. If true, API updates the view count in the given channel as well as the total view count.

boolean `isUrlName`: Optional, defaults to false. If true, indicates that the last portion of the endpoint is a URL name instead of an article ID. Available in API v44.0 and later

Output:

The detailed fields of the article, if the article is online and visible to the current user.

- **Article Detail**

  Full detail of an article, with complete metadata and layout-driven fields used for display of an article. It includes all the same properties as an Article Summary representation.

```
{
         "id":  Id,    // articleId,
         "articleNumber": String,
         "articleType": String,  // apiName of the article type, available in API
v44.0 and later
         "title":  String,
         "urlName": String,   // available in API v44.0 and later
         "summary":  String,
         "url":  URL,
         "versionNumber": Int,
         "createdDate": Date,  // in ISO8601 format
         "createdBy": User Summary on page 272,
         "lastModifiedDate": Date,   // in ISO8601 format
         "lastModifiedBy": User Summary on page 272,
         "lastPublishedDate": Date, // in ISO8601 format
        "layoutItems": [ Article Field, ... ],  // standard and custom fields visible
 to the user, sorted based on the layouts of the article type.
         "categories": [ Data Category Groups, ... ],
         "appUpVoteCount": Int,
         "cspUpVoteCount": Int,
         "prmUpVoteCount": Int,
         "pkbUpVoteCount": Int,
         "appDownVoteCount": Int,
         "cspDownVoteCount": Int,
         "prmDownVoteCount": Int,
         "pkbDownVoteCount": Int,
         "allViewCount": Int,
         "appViewCount": Int,
         "cspViewCount": Int,
         "prmViewCount": Int,
         "pkbViewCount": Int,
         "allViewScore": Double,
         "appViewScore": Double,
         "cspViewScore": Double,
         "prmViewScore": Double,
         "pkbViewScore": Double
         }
```

- **User Summary**

```
{
        "id": String
        "isActive": boolean   // true/false
        "userName": String   // login name
        "firstName": String
        "lastName": String
        "email": String
        "url": String              // to the chatter user detail url:
/services/data/vXX.X/chatter/users/{userId}, for guest user, it will return null.
        }
```

- **Article Field**

An individual field of article information, which is listed in an Article Detail in the order required by the administrator's layout.

```
{
        "type":  Enum,          // see the Notes
        "name": String,         // In API v43.0 and earlier, the developer name. In
API v44.0 and later, the API name.
        "label": String,          // label
        "value": String,
        }
```

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/support/knowledgeArticles/kA0xx000000000LCAQ

       HTTP Headers:
       Content-Type: application/json; charset=UTF-8
       Accept: application/json
       Accept-Language: en-US
```

Example Response Body

```
{
       "allViewCount" : 17,
       "allViewScore" : 100.0,
       "appDownVoteCount" : 0,
       "appUpVoteCount" : 0,
       "appViewCount" : 17,
       "appViewScore" : 100.0,
       "articleNumber" : "000001004",
       "categoryGroups" : [ ],
       "createdBy" : {
       "email" : "user@company.com",
       "firstName" : "Test",
       "id" : "005xx000001SvoMAAS",
       "isActive" : true,
       "lastName" : "User",
       "url" : "/services/data/v60.0/chatter/users/005xx000001SvoMAAS",
```

```
"userName" : "admin@salesforce.org"
},
"createdDate" : "2016-06-21T21:10:54Z",
"cspDownVoteCount" : 0,
"cspUpVoteCount" : 0,
"cspViewCount" : 0,
"cspViewScore" : 0.0,
"id" : "kA0xx000000000LCAQ",
"lastModifiedBy" : {
"email" : "user@company.com",
"firstName" : "Test",
"id" : "005xx000001SvoMAAS",
"isActive" : true,
"lastName" : "User",
"url" : "/services/data/v60.0/chatter/users/005xx000001SvoMAAS",
"userName" : "admin@salesforce.org"
},
"lastModifiedDate" : "2016-06-21T21:11:02Z",
"lastPublishedDate" : "2016-06-21T21:11:02Z",
"layoutItems" : [ {
"label" : "Out of Date",
"name" : "IsOutOfDate",
"type" : "CHECKBOX",
"value" : "false"
}, {
"label" : "sample",
"name" : "sample",
"type" : "PICK_LIST",
"value" : null
}, {
"label" : "Language",
"name" : "Language",
"type" : "PICK_LIST",
"value" : "en_US"
}, {
"label" : "MyNumber",
"name" : "MyNumber",
"type" : "NUMBER",
"value" : null
}, {
"label" : "My File",
"name" : "My_File",
"type" : "FILE",
"value" : null
} ],
"pkbDownVoteCount" : 0,
"pkbUpVoteCount" : 0,
"pkbViewCount" : 0,
"pkbViewScore" : 0.0,
"prmDownVoteCount" : 0,
"prmUpVoteCount" : 0,
"prmViewCount" : 0,
"prmViewScore" : 0.0,
"summary" : "The number of characters required for complete coverage of all these
```

```
languages' needs cannot fit in the 256-character code space of 8-bit character encodings,
requiring at least a 16-bit fixed width encoding or multi-byte variable-length encodings.
\r\n\r\nAlthough CJK encodings have common character sets, the encodings often used to
represent them have been developed separately by different East Asian governments and
software companies, and are mutually incompatible. Unicode has attempted, with some
controversy, to unify the character sets in a process known as Han unification.\r\n\r\nCJK
character encodings should consist minimally of Han characters p",
      "title" : "Test Images",
      "url" : "/services/data/v60.0/support/knowledgeArticles/kA0xx000000000LCAQ",
      "versionNumber" : 7
      }
```

Usage

Parameterized Search

Executes a simple REST search using parameters instead of a SOSL clause. Indicate parameters in the URI with the GET method. Or, use the POST method to create complex searches in a request body.

Search with Parameters in the URI

Get search results using simple URI parameters instead of using SOSL. Make basic queries without defining a large SOSL query. Use this API when you have a basic use case to cover, replacing FIND `searchString` IN ALL FIELDS by just including the search string in the URI. This resource is available in REST API version 36.0 and later.

Syntax

URI

`/services/data/vXX.X/parameterizedSearch/?q=searchString`

Formats

JSON, XML

HTTP methods

GET

Authentication

`Authorization: Bearer token`

Required Global Parameters

Name	Description
q	A search string that is properly URL-encoded. Note: SOSL clauses aren't supported. Available in version 36.0 and later.

Optional Global Parameters

Name	Type	Description
dataCategory	string	Single value. If an organization uses Salesforce Knowledge articles or answers, `dataCategory` filters all search results based on one data category. For example, `dataCategory=GlobalCategory__c below NorthAmerica__c`. When using `dataCategories`, specify a Salesforce Knowledge article or answer type with `sobject` and all the required parameters. For example: `q=tourism&sobject=KnowledgeArticleVersion&KnowledgeArticleVersion.where=language='en_US'+and+publishStatus='online'&KnowledgeArticleVersion.fields=id,title&dataCategory=Location__c+Below+North_America__c` If you require multiple `dataCategory` filters, use `dataCategories` with the POST method.
defaultLimit	string	Single value. The maximum number of results to return for each `sobject` (GET) or `sobjects` (POST) specified. The maximum `defaultLimit` is 2000. At least one `sobject` must be specified. GET example: `defaultLimit=10&sobject=Account&sobject=Contact`. When an `sobject` limit is specified using `sobject.limit=value`, such as `Account.limit=10`, this parameter is ignored for that object.
division	string	Single value. Filters search results based on the division field. For example in the GET method, `division=global`. Specify a division by its name rather than ID. All searches within a specific division also include the `global` division.
fields	string	Comma-separated list of one or more fields to return in the response for each `sobject` specified. At least one `sobject` must be specified at the global level. For example: `fields=id&sobject=Account&sobject=Contact`. The global `fields` parameter is overridden when `sobject` are specified using *`sobject`*`.fields=`*`field names`*. For example, `Contact.fields=id,FirstName,LastName` would override the global setting of just returning the `id`. If unspecified, then the search results contain the IDs of records matching all fields for the specified object. **Functions** The following optional functions can be used within the `fields` parameter. • `toLabel`: Translates response field value into the user's language. For example, `Lead.fields=id,toLabel(Status)`. This function requires extra setup.

Name	Type	Description
		• `convertCurrency`: Converts response currency fields to the user's currency. For example, `Opportunity.fields=id,convertCurrency(Amount)`. This function requires extra setup. Multi-currency must be enabled in your org. • `format`: Applies localized formatting to standard and custom number, date, time, and currency fields. For example, `Opportunity.fields=id,format(Amount)`. Aliasing is supported in `fields` for `toLabel`, `convertCurrency`, and `format`. In addition, aliasing is required when the query includes the same field multiple times. For example, `Opportunity.fields=id,format(Amount) AliasAmount`
`in`	string	Scope of fields to search. If you specify one or more scope values, the fields are returned for all found objects. Use one of the following values: • `ALL` • `NAME` • `EMAIL` • `PHONE` • `SIDEBAR` This clause doesn't apply to articles, documents, feed comments, feed items, files, products, and solutions. If any of these objects are specified, the search isn't limited to specific fields; all fields are searched.
`metadata`	string	Specifies if metadata should be returned in the response. No metadata is returned by default. To include metadata in the response, use the `LABELS` value, which returns the display label for the fields returned in search results. For example: `?q=Acme&metadata=LABELS`
`netWorkIds`	string	Filters search results by a comma-separated list. A network ID represents the Experience Cloud site ID.
`offset`	string	Single value. The starting row offset into the result set returned. The maximum `offset` is 2000. Only one `sobject` can be specified when using this parameter.
`overallLimit`	string	Single value. The maximum number of results to return across all `sobject` parameters specified. The maximum `overallLimit` is 2000.
`pricebookId`	string	Single value. Filters product search results by a price book ID for only the Product2 object. The price book ID must be associated with the product that you're searching for. For example, `?q=laptop&sobject=product2&pricebookId=01sxx0000002MffAAE`

Name	Type	Description
`snippet`	string	The target length (maximum number of snippet characters) to return in Salesforce Knowledge article, case, case comment, feed, feed comment, idea, and idea comment search results. The `snippet` parameter displays contextual excerpts and highlights the search term for each article in the search results. Snippet results are used to differentiate matches to the search term in article search results. The target length can be from 50 to 1000 characters. Snippet and highlights are generated from email, text, and text area (long and rich) fields. Snippets aren't displayed for partially matching searches or if the user doesn't have access to the field that contains the snippet. Snippets are only displayed when 20 or fewer results are returned on a page. At least one of the following `sobject` values must be specified. • To search a specific article type, use the article type name with the suffix `__kav`. • To search all article types, use `KnowledgeArticleVersion`. • To search case, case comment, feed, feed comment, idea, and idea comment types, use `Case`, `CaseComment`, `FeedItem`, `FeedComment`, `Idea`, and `IdeaComment`. For example, `q=tourism&sobject=Case&snippet=500`.
`sobject`	string	Objects to return in the response. Must be a valid object type. You can use multiple `sobject` values, such as `sobject=Account&sobject=Contact`. If unspecified, then the search results contain the IDs of all objects.
`spellCorrection`	boolean	Specifies whether spell correction is enabled for a user's search. When set to `true`, spell correction is enabled for searches that support spell correction. The default value is `true`. For example: `q=Acme&sobject=Account&Account.fields=id&spellCorrection=true`
`updateTracking`	string	Specifies a value of `true` to track keywords that are used in Salesforce Knowledge article searches only. If unspecified, the default value of `false` is applied.
`updateViewStat`	string	Specifies a value of `true` to update an article's view statistics. Valid only for Salesforce Knowledge article searches. If unspecified, the default value of `false` is applied.

`sobject`-level Parameters

The following optional parameters can be used with the `sobject` parameter in a GET method to further refine search results. These settings would override any settings specified at the global level.

The format is *`sobject.parameter`*, such as `Account.fields`. An `sobject` must be specified to use these parameters, for example, `sobject=Account&Account.fields=id,name`.

Name	Type	Description
`fields`	string	Comma-separated list of one or more fields to return in the response. For example, `KnowledgeArticleVersion.fields=id,title`.
`limit`	string	Specifies the maximum number of rows that are returned for the `sobject`. For example, `Account.limit=10`.
`orderBy`	string	Controls the field order of the results using the following syntax `orderBy = field {ASC|DESC} [NULLS_{FIRST|LAST}]` For example: `Account.orderBy=Name` • `ASC`: ascending. Default. • `DESC`: descending. • `NULLS_FIRST`: Null records at the beginning of the results. Default. • `NULLS_LAST`: Null records at the end of the results.
`where`	string	Filter search results for this object by specific field values. For example, `Account.where = conditionExpression`. Here the `conditionExpression` of the WHERE clause uses the following syntax: `fieldExpression [logicalOperator fieldExpression2 ... ]`. Add multiple field expressions to a condition expression by using logical and comparison operators. For example, `KnowledgeArticleVersion.where=publishstatus='online' and language='en_US'`.

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/parameterizedSearch/?q=Acme&sobject=Account&Account.fields=id,name&Account.limit=10
```

Search with Parameters in the Request Body

Search by defining parameters in the request body Access advanced search that offers more control over how the search query executes. It also allows you to filter using several DataCategories, networks (sites), orderBy constraints, and filters. This resource is available in REST API version 36.0 and later.

Syntax

URI

`/services/data/vXX.X/parameterizedSearch/`

Formats

JSON, XML

HTTP methods

POST

Authentication

`Authorization: Bearer token`

Required Global Parameters

Name	Description
`q`	A search string that is properly URL-encoded. Note: SOSL clauses aren't supported. Available in version 36.0 and later.

Optional Global Parameters

Name	Type	Description

`dataCategories`	dataCategoriesFilter[]	If an organization uses Salesforce Knowledge articles or answers, filter all search results based on one or more data categories.
		When using `dataCategories`, specify a Salesforce Knowledge article or answer type with `sobjects` and the required parameters.
		For example:
		`{`
		`  "q":"Acme",`
		`  "fields":["id", "title"],`
		`  "sobjects":[{"name":"KnowledgeArticleVersion",`
		`"where":"language='en_US' and publishstatus='draft'"}],`
		`  "dataCategories":[`
		`    {"groupName" : "location__c", "operator":"below",`
		`"categories":["North_America__c"]}`
		`                  ]`
		`}`

`defaultLimit`	string	Single value. The maximum number of results to return for each `sobject` (GET) or `sobjects` (POST) specified. The maximum `defaultLimit` is 2000. At least one `sobject` must be specified. GET example: `defaultLimit=10&sobject=Account&sobject=Contact`. When an `sobject` limit is specified using `sobject.limit=value`, such as `Account.limit=10`, this parameter is ignored for that object.
`division`	string	Single value. Filters search results based on the division field. For example in the GET method, `division=global`. Specify a division by its name rather than ID. All searches within a specific division also include the `global` division.

<table>
<tr><th>Name</th><th>Type</th><th>Description</th></tr>
<tr><td><code>fields</code></td><td>string[]</td><td>

Array of one or more fields to return in the response for each `sobjects` specified. At least one `sobjects` must be specified at the global level.

For example:

```
{
  "q":"Acme",
  "fields":["Id", "Name", "Phone"],
  "sobjects":[{"name": "Account"},
              {"name": "Contact", "fields":["Id",
"FirstName", "LastName"]},
              {"name": "Lead"}]
}
```

The global `fields` parameter is overridden when `sobjectsFilter[]` fields are specified. Such as, in the previous example, `Id`, `FirstName`, and `LastName` is returned for `Contact` instead of the global fields of `Id`, `Name` and `Phone`.

If unspecified, then the search results contain the IDs of records matching all fields for the specified object.

Functions

The following optional functions can be used within the `fields` parameter.

- `toLabel`: Translates response field value into the user's language. This function requires extra setup. For example:

```
{
...
  "sobjects":[ {"name": "Lead", "fields":["Id",
"toLabel(Status)"]},
...
}
```

- `convertCurrency`: Converts response currency fields to the user's currency. This function requires extra setup. Multi-currency must be enabled in the org. For example:

```
{
...
  "sobjects":[ {"name": "Opportunity", "fields":["Id",
"convertCurrency(Amount)"]}]
...
}
```

- `format`: Applies localized formatting to standard and custom number, date, time, and currency fields. For example:

```
{
...
  "sobjects":[ {"name": "Opportunity", "fields":["Id",
"format(Amount)"]}]
...
}
```

</td></tr>
</table>

<table>
<tr><th>Name</th><th>Type</th><th>Description</th></tr>
<tr><td></td><td></td><td>Aliasing is supported within <code>fields</code> for <code>toLabel</code>, <code>convertCurrency</code>, and <code>format</code>. In addition, aliasing is required when the query includes the same field multiple times. For example:
<pre>{
...
  "subjects":[{"name": "Opportunity", "fields":["Id",
"format(Amount) AliasAmount"]}]
...
}</pre></td></tr>
<tr><td><code>in</code></td><td>string</td><td>Scope of fields to search. If you specify one or more scope values, the fields are returned for all found objects.
Use one of the following values:
<code>ALL</code>
<code>NAME</code>
<code>EMAIL</code>
<code>PHONE</code>
<code>SIDEBAR</code>
This clause doesn't apply to articles, documents, feed comments, feed items, files, products, and solutions. If any of these objects are specified, the search isn't limited to specific fields; all fields are searched.</td></tr>
<tr><td><code>metadata</code></td><td>string</td><td>Specifies if metadata should be returned in the response. No metadata is returned by default. To include metadata in the response, use the <code>LABELS</code> value, which returns the display label for the fields returned in search results. For example: <code>?q=Acme&metadata=LABELS</code></td></tr>
<tr><td><code>netWorkIds</code></td><td>string[]</td><td>Filters search results by an array.
A network ID represents the Experience Cloud site ID.</td></tr>
<tr><td><code>offset</code></td><td>string</td><td>Single value. The starting row offset into the result set returned.
The maximum <code>offset</code> is 2000.
Only one <code>sobject</code> can be specified when using this parameter.</td></tr>
<tr><td><code>overallLimit</code></td><td>string</td><td>Single value. The maximum number of results to return across all <code>sobject</code> parameters specified.
The maximum <code>overallLimit</code> is 2000.</td></tr>
<tr><td><code>pricebookId</code></td><td>string</td><td>Single value. Filters product search results by a price book ID for only the Product2 object. The price book ID must be associated with the product that you're searching for. For example, <code>?q=laptop&sobject=product2&pricebookId=01sxx0000002MffAAE</code></td></tr>
<tr><td><code>snippet</code></td><td>string</td><td>The target length (maximum number of snippet characters) to return in Salesforce Knowledge article, case, case comment, feed, feed comment, idea, and idea comment</td></tr>
</table>

<table>
<tr><th>Name</th><th>Type</th><th>Description</th></tr>
<tr><td></td><td></td><td>search results. The <code>snippet</code> parameter displays contextual excerpts and highlights the search term for each article in the search results. Snippet results are used to differentiate matches to the search term in article search results. The target length can be from 50 to 1000 characters.

Snippet and highlights are generated from email, text, and text area (long and rich) fields. Snippets aren't displayed for partially matching searches or if the user doesn't have access to the field that contains the snippet. Snippets are only displayed when 20 or fewer results are returned on a page.

At least one of the following <code>sobject</code> values must be specified.
To search a specific article type, use the article type name with the suffix <code>__kav</code>.
To search all article types, use <code>KnowledgeArticleVersion</code>.
To search case, case comment, feed, feed comment, idea, and idea comment types, use <code>Case</code>, <code>CaseComment</code>, <code>FeedItem</code>, <code>FeedComment</code>, <code>Idea</code>, and <code>IdeaComment</code>.

For example, <code>q=tourism&sobject=Case&snippet=500</code>.</td></tr>
<tr><td><code>sobjects</code></td><td>sobjectsFilter[]</td><td>Objects to return in the response. Must contain valid object types. Use with the required parameters.

For example:<pre>{
  "q":"Acme",
  "fields":["id", "title"],
  "sobjects":[{"name":"Solution__kav",
"where":"language='en_US' and publishstatus='draft'"},
            {"name":"FAQ__kav", "where":"language='en_US'
and publishstatus='draft'"}]
}</pre>If unspecified, then the search results contain the IDs of all objects.</td></tr>
<tr><td><code>spellCorrection</code></td><td>boolean</td><td>Specifies whether spell correction is enabled for a user's search. When set to <code>true</code>, spell correction is enabled for searches that support spell correction. The default value is <code>true</code>.

For example:
<code>q=Acme&sobject=Account&Account.fields=id&spellCorrection=true</code></td></tr>
<tr><td><code>updateTracking</code></td><td>string</td><td>Specifies a value of <code>true</code> to track keywords that are used in Salesforce Knowledge article searches only.

If unspecified, the default value of <code>false</code> is applied.</td></tr>
<tr><td><code>updateViewStat</code></td><td>string</td><td>Specifies a value of <code>true</code> to update an article's view statistics. Valid only for Salesforce Knowledge article searches.

If unspecified, the default value of <code>false</code> is applied.</td></tr>
</table>

`dataCategoriesFilter[]` Parameters

Parameters must be specified in the order presented in the table (`groupName`, `operator`, `categories`).

Name	Type	Description
`groupName`	string	The name of the data category group to filter by.
`operator`	string	Valid values: • `ABOVE` • `ABOVE_OR_BELOW` • `AT` • `BELOW`
`categories`	string[]	The name of the categories to filter by.

`sobjectsFilter[]` Parameters

Name	Type	Description
`fields`	string[]	Array of one or more fields to return in the response for the `sobject`.
`limit`	string	Specify the maximum number of rows that are returned for the `sobject`.
`name`	string	Name of the `sobject` to return in the response.

`orderBy`	string	Controls the field order of the results using the following syntax `"orderBy" : "field {ASC|DESC} [NULLS_{FIRST|LAST}]"`
		For example:
		`{`
		`...`
		`"sobjects":[ {"name": "Account", "fields":["Id", "name"], "orderBy": "Name DESC Nulls_last"}]`
		`...`
		`}`
		• `ASC`: ascending. Default.
		• `DESC`: descending.
		• `NULLS_FIRST`: Null records at the beginning of the results. Default.
		• `NULLS_LAST`: Null records at the end of the results.

`where`	string	Filter search results for this object by specific field values. For example, `where : conditionExpression`. Here the `conditionExpression` of the WHERE clause uses the following syntax: `fieldExpression [logicalOperator fieldExpression2 ... ]`. Add multiple field expressions to a condition expression by using logical and comparison operators.

Example

Example Request Body

```
{
  "q":"Smith",
  "fields" : ["id", "firstName", "lastName"],
  "sobjects":[{"fields":["id", "NumberOfEmployees"],
         "name": "Account",
         "limit":20},
        {"name": "Contact"}],
  "in":"ALL",
  "overallLimit":100,
  "defaultLimit":10
}
```

Portability

The Portability resource compiles customer information across objects and fields that you identified when creating a portability policy in Salesforce Privacy Center. You can locate your customers' personally identifiable information (PII) across multiple records when using REST API version 50.0 and later. Data portability satisfies your customers' right to obtain a copy of their PII that is kept in your organization's platform. To meet privacy regulations, such as the General Data Protection Regulation (GDPR), data portability requests must be fulfilled within one month of when the request is made.

Compile Data for a Portability Request

Aggregate your data subject's personally identifiable information (PII) into one file using the POST method of the Portability resource. The PII includes data found in the Account, Contact, Individual, Lead, Person, and User objects. You receive a response with a URL to download the file, a policy file ID, and information on the objects and fields you selected when creating the policy. Use the policy file ID to execute the Portability resource with the GET method. This resource is available in REST API version 50.0 and later.

To use the Portability resource, you must have the ModifyAllData or PrivacyDataAccess user permission. Ensure that your Salesforce admin has granted you these permissions.

Syntax

URI

`/services/data/vXX.X/consent/dsr/rtp/execute`

Formats

JSON

HTTP methods

POST

Authentication

`Authorization: Bearer token`

Request body

```
{
```

```
   “dataSubjectId”:”<root ID>”,
   “policyName”:”<policyName>”
}
```

Request parameters

Parameter	Description
dataSubjectId	The ID of the customer making the request. The ID is 15 or 18 characters long, and found in the Account, Contact, Individual, Lead, Person, and User objects.
policyName	The name of an active policy. This contains the object in the dataSubjectId parameter.

Example

Example Request

```
curl -X POST
https://MyDomainName.my.salesforce.com/services/data/v60.0/consent/dsr/rtp/execute -H
"Authorization: Bearer token" -H "Content-Type: application/json" -d
"@exampleRequestBody.json"
```

Example Response Body

```
{
   “status" : "SUCCESS",
   "warnings" : [ ],
   "result" : {
        "policyFileStatus" : "In Progress",
        "policyFileUrl" :
"https://MyDomainName.my.salesforce.com/servlet/policyFileDownload?file=0jeS70000004CBO",

        "policyFileId" : "0jeS70000004CBO"
   }
}
```

Get the Status of Your Portability Request

See the status of your Portability POST request by using a Portability GET request. Use the policy file ID from the POST method response to execute the GET method. This resource is available in REST API version 50.0 and later.

To use the Portability API, you must have the ModifyAllData or PrivacyDataAccess user permission. Ensure that your Salesforce admin has granted you these permissions.

The response body contains this information:

Value	Description
policyFileStatus	The status of the file being compiled. Values are: In Progress, Complete, or Failed.
policyFileURL	The URL to a servlet, where you download the file after it's compiled.

Value	Description
`policyFileID`	The ID of the file being compiled, which is returned in the POST method response. The ID is 15 characters.

Note: Starting with the Spring '21 release, Privacy Center automatically deletes files generated by Portability API after 60 days. You receive a reminder seven days before a file is deleted.

Syntax

URI

`/services/data/vXX.X/consent/dsr/rtp/execute`

Formats

JSON

HTTP methods

GET

Authentication

`Authorization: Bearer token`

Request body

None

Request parameters

Parameter	Description
`policyFileId`	The ID of the file being compiled, which is returned in the POST method response. The ID is 15 characters.

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/consent/dsr/rtp/execute?policyFileId=0jeS70000004CBO
 -H "Authorization: Bearer token"
```

Example Response Body

```
{
   “status" : "SUCCESS",
   "warnings" : [ ],
   "result" : {
       "policyFileStatus" : "Failed",
       "policyFileUrl" :
"https://MyDomainName.my.salesforce.com/servlet/policyFileDownload?file=0jeS70000004CBO",

       "policyFileId" : "0jeS70000004CBO"
```

```
    }
}
```

Process Approvals

Accesses all approval processes. Can also be used to submit a particular record if that entity supports an approval process and one has already been defined. Records can be approved and rejected if the current user is an assigned approver.

IN THIS SECTION:

Get Process Approvals

Gets a list of all approval processes. This resource is available in REST API version 30.0 and later.

Syntax

URI
`/services/data/vXX.X/process/approvals/`

Formats
JSON, XML

HTTP methods
GET

Authentication
`Authorization: Bearer token`

Request parameters
None required

Example
See Get a List of All Approval Processes..

Submit, Approve, or Reject Process Approvals

Submits a particular record if that entity supports an approval process and one has already been defined. Records can be approved and rejected if the current user is an assigned approver. This resource is available in REST API version 30.0 and later.

When using a POST request to do bulk approvals, the requests that succeed are committed and the requests that don't succeed send back an error.

Syntax

URI

/services/data/v***XX.X***/process/approvals/

Formats

JSON, XML

HTTP methods

POST

Authentication

Authorization: Bearer ***token***

Request parameters

None required

Request body

The request body contains an array of process requests that contain the following information:

Name	Type	Description
actionType	string	Represents the kind of action to take: *Submit*, *Approve*, or *Reject*.
contextActorId	ID	The ID of the submitter who's requesting the approval record.
contextId	ID	The ID of the item that is being acted upon.
comments	string	The comment to add to the history step associated with this request.
nextApproverIds	ID[]	If the process requires specification of the next approval, the ID of the user to be assigned the next request.
processDefinitionNameOrId	string	The developer name or ID of the process definition.
skipEntryCriteria	boolean	Determines whether to evaluate the entry criteria for the process (true) or not (false) if the process definition name or ID isn't null. If the process definition name or ID isn't specified, this argument is ignored, and standard evaluation is followed based on process order. By default, the entry criteria isn't skipped if it's not set by this request.

Response body

The response contains an array of process results that contain the following information:

Name	Type	Description
actorIds	ID[]	IDs of the users who are currently assigned to this approval step.
entityId	ID	The object being processed.
errors	Error[]	The set of errors returned if the request failed.
instanceId	ID	The ID of the ProcessInstance associated with the object submitted for processing.

Name	Type	Description
instanceStatus	string	The status of the current process instance (not an individual object but the entire process instance). The valid values are “Approved,” “Rejected,” “Removed,” or “Pending.”
newWorkItemIds	ID[]	Case-insensitive IDs that point to ProcessInstanceWorkitem items (the set of pending approval requests)
success	boolean	`true` if processing or approval completed successfully.

Example

- See Submit a Record for Approval.
- See Approve a Record.
- See Reject a Record.
- See Bulk Approvals.

Return HTTP Headers for Process Approvals

Returns only the headers that are returned by sending a GET request to the process approvals resource. This gives you a chance to see returned header values of the GET request before retrieving the content. This resource is available in REST API version 30.0 and later.

Syntax

URI

`/services/data/vXX.X/process/approvals/`

Formats

JSON, XML

HTTP methods

HEAD

Authentication

`Authorization: Bearer token`

Request parameters

None required

Example

Example Request

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/process/approvals/ -H
"Authorization: Bearer token"
```

Example Response Body

```
HTTP/1.1 200 OK
Date: Mon, 21 Nov 2022 22:56:26 GMT
```

Process Rules

Accesses a list of all active workflow rules. Use the GET method to retrieve records or fields. Use the HEAD method to retrieve information in HTTP headers. Use the POST method to trigger all active workflow rules.

To access all workflow rules that are associated with a specific sObject, use the Process Rule List for an sObject resource. To access a specific workflow rule that is associated with a specific sObject, use the Process Rule for an sObject resource.

Cross-object workflow rules can't be invoked using REST API.

IN THIS SECTION:

Get Process Rules

Gets all active workflow rules. This resource is available in REST API version 30.0 and later.

Syntax

URI

`/services/data/vXX.X/process/rules/`

Formats

JSON, XML

HTTP methods

GET

Authentication

`Authorization: Bearer token`

Request parameters

None required

Example

See Get a List of Process Rules.

Trigger Process Rules

Triggers all active workflow rules. All rules associated with the specified ID are evaluated, regardless of the evaluation criteria. All IDs must be for records on the same object. This resource is available in REST API version 30.0 and later.

Syntax

URI

`/services/data/vXX.X/process/rules/`

Formats

JSON, XML

HTTP methods

POST

Authentication

`Authorization: Bearer token`

Request parameters

None required

Request body

The request body contains an array of context IDs:

Name	Type	Description
`contextIds`	ID[]	An array of IDs of the items that are being acted upon.

Example

See Trigger Process Rules.

Return HTTP Headers for Process Rules

Returns only the headers that are returned by sending a GET request to the process rules resource. This gives you a chance to see returned header values of the GET request before retrieving the content. This resource is available in REST API version 30.0 and later.

Syntax

URI

`/services/data/vXX.X/process/rules/`

Formats

JSON, XML

HTTP methods

HEAD

Authentication

`Authorization: Bearer token`

Request parameters

None required

Example

Example Request

```
curl -X HEAD --head
https://MyDomainName.my.salesforce.com/services/data/v60.0/process/rules/ -H
"Authorization: Bearer token"
```

Example Response Body

```
HTTP/1.1 200 OK
Date: Mon, 21 Nov 2022 22:56:26 GMT
```

Process Rule for an sObject

Accesses an active workflow rule for an sObject. Use the GET method to retrieve the record or fields. Use the HEAD method to retrieve information in HTTP headers. Use the POST method to trigger the workflow rule.

Cross-object workflow rules can't be invoked using REST API.

To access all workflow rules, use the Process Rules resource. To access a specific workflow rule that is associated with a specific sObject, use the Process Rule List for an sObject resource.

IN THIS SECTION:

Get a Process Rule for an sObject

Gets the fields of a specific workflow rule for a specific sObject.

Cross-object workflow rules can't be invoked using REST API.

URI

`/services/data/vXX.X/process/rules/sObjectName/workflowRuleId`

Available since release

30.0

Formats

JSON, XML

HTTP methods

GET

Authentication

`Authorization: Bearer token`

Request parameters
None required

Example usage

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/process/rules/Account/01QD0000000APli
 -H "Authorization: Bearer token"
```

Example request body
None required

Example JSON response body

```
{
  "actions" : [ {
    "id" : "01VD0000000D2w7",
    "name" : "ApprovalProcessTask",
    "type" : "Task"
    } ],
    "description" : null,
    "id" : "01QD0000000APli",
    "name" : "My Rule",
    "namespacePrefix" : null,
    "object" : "Account"
}
```

Trigger a Process Rule for an sObject

Triggers an active workflow rule regardless of the evaluation criteria.

URI
`/services/data/vXX.X/process/rules/sObjectName/workflowRuleId`

Available since release
30.0

Formats
JSON, XML

HTTP methods
POST

Authentication
`Authorization: Bearer token`

Request parameters
None required

Request body
None required

Example usage

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/process/rules/Account/01XXX000000XxxXXX
 -H "Authorization: Bearer token"
```

Example JSON response body

```
{
  "errors" : null,
  "success" : true
}
```

Return HTTP Headers for a Process Rule of an sObject

Returns only the headers that are returned by sending a GET request to the process rules resource for a specific process rule of an sObject. This gives you a chance to see returned header values of the GET request before retrieving the content.

URI

`/services/data/vXX.X/process/rules/sObjectName/workflowRuleId`

Available since release

30.0

Formats

JSON, XML

HTTP methods

HEAD

Authentication

`Authorization: Bearer token`

Request parameters

None required

Example usage

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/process/rules/Account/01XXX000000/
 -H "Authorization: Bearer token"
```

Example request body

None required

Example response body

```
HTTP/1.1 200 OK
Date: Mon, 21 Nov 2022 22:56:26 GMT
```

Process Rule List for an sObject

Accesses a list of all active workflow rules for an sObject. Use the GET method to retrieve records or fields. Use the HEAD method to retrieve information in HTTP headers.

Cross-object workflow rules can’t be invoked using REST API.

To access all workflow rules, use the Process Rules resource. To access a specific workflow rule that is associated with a specific sObject, use the Process Rule for an sObject resource.

IN THIS SECTION:

Get Process Rules for an sObject

Gets all active workflow rules for an sObject. This resource is available in REST API version 30.0 and later.

Syntax

URI

`/services/data/vXX.X/process/rules/sObject`

Formats

JSON, XML

HTTP methods

GET

Authentication

`Authorization: Bearer token`

Request parameters

None required

Request body

None required

Example

Example Request

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/process/rules/Account
-H "Authorization: Bearer token"
```

Example Response Body

```
{
  "rules" : {
    "Account" : [ {
      "actions" : [ {
        "id" : "01VD0000000D2w7",
        "name" : "ApprovalProcessTask",
        "type" : "Task"
      } ],
      "description" : null,
      "id" : "01QD0000000APli",
      "name" : "My Rule",
```

```
      "namespacePrefix" : null,
      "object" : "Account"
    } ]
  }
}
```

Return HTTP Headers for Process Rules of an sObject

Returns only the headers that are returned by sending a GET request to the process rules resource for all process rules of an sObject. This gives you a chance to see returned header values of the GET request before retrieving the content. This resource is available in REST API version 30.0 and later.

Syntax

URI

`/services/data/vXX.X/process/rules/sObject`

Formats

JSON, XML

HTTP methods

HEAD

Authentication

`Authorization: Bearer token`

Request parameters

None required

Request body

None required

Example

Example Request

```
curl -X HEAD --head
https://MyDomainName.my.salesforce.com/services/data/v60.0/process/rules/Account/ -H
"Authorization: Bearer token"
```

Example Response Body

```
HTTP/1.1 200 OK
Date: Mon, 21 Nov 2022 22:56:26 GMT
```

Product Schedules

Work with revenue and quantity schedules for opportunity products. Establish or reestablish a product schedule with multiple installments for an opportunity product. Delete all installments in a schedule.

This resource is available in REST API version 43.0 and later. In API version 46.0 and later, established and re-established schedules support custom fields, validation rules, and Apex triggers.

IN THIS SECTION:

Get Product Schedules

Get revenue and quantity schedules for opportunity products. This resource is available in REST API version 43.0 and later.

Syntax

URI

`/services/data/vXX.X/sobjects/OpportunityLineItem/OpportunityLineItemId/OpportunityLineItemSchedules`

Formats

JSON, XML

HTTP methods

GET

Authentication

`Authorization: Bearer token`

Request body

`None`

Request parameters

`None`

Create Product Schedules

Establish or reestablish a product schedule with multiple installments for an opportunity product. This resource is available in REST API version 43.0 and later. In API version 46.0 and later, established and re-established schedules support custom fields, validation rules, and Apex triggers.

Syntax

URI

`/services/data/vXX.X/sobjects/OpportunityLineItem/OpportunityLineItemId/OpportunityLineItemSchedules`

Formats

JSON, XML

HTTP methods

PUT

Authentication

`Authorization: Bearer token`

Request parameters

Parameter	Description
`type`	The type of the schedule. Required when establishing OpportunityLineItemSchedules. Valid values include `Quantity`, `Revenue`, or `Both`.
`quantity`	The total number of units to be repeated or divided in a quantity schedule. Must be an integer other than 0.
`quantityScheduleType`	The type of the quantity schedule, if the product has one. Valid values are `Divide` or `Repeat`.
`quantityScheduleInstallmentPeriod`	If the product has a quantity schedule, the amount of time covered by the schedule. Valid values are `Daily`, `Weekly`, `Monthly`, `Quarterly`, or `Yearly`.
`quantityScheduleInstallmentsNumber`	If the product has a quantity schedule, the number of installments. Can be an integer from 1 to 150.
`quantityScheduleStartDate`	The date the quantity schedule starts. Format is YYYY-MM-DD.
`revenue`	The amount of revenue to be repeated or divided.
`revenueScheduleType`	The type of the revenue schedule, if the product has one. Valid values are `Divide` or `Repeat`.
`revenueScheduleInstallmentPeriod`	If the product has a revenue schedule, the amount of time covered by the schedule. Valid values are `Daily`, `Weekly`, `Monthly`, `Quarterly`, or `Yearly`.
`revenueScheduleInstallmentsNumber`	If the product has a revenue schedule, the number of installments. Can be an integer from 1 to 150.
`revenueScheduleStartDate`	The date the revenue schedule starts. Format is YYYY-MM-DD.

Examples

Establish both quantity and revenue schedules for an opportunity product.

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/OpportunityLineItem/00kR0000001WJJAIA4/OpportunityLineItemSchedules
 -H "Authorization: Bearer token"
```

JSON Request body

```
{
"type": "Both",
"quantity": 100,
"quantityScheduleType": "Repeat",
"quantityScheduleInstallmentPeriod": "Monthly",
```

```
"quantityScheduleInstallmentsNumber": 12,
"quantityScheduleStartDate": "2018-09-15",
"revenue": 100,
"revenueScheduleType": "Repeat",
"revenueScheduleInstallmentPeriod": "Monthly",
"revenueScheduleInstallmentsNumber": 12,
"revenueScheduleStartDate": "2018-09-15"
}
```

Establish a revenue schedule only for an opportunity product.

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/OpportunityLineItem/00kR0000001WJJAIA4/OpportunityLineItemSchedules
 -H "Authorization: Bearer token"
```

JSON Request body

```
{
"type": “Revenue”,
"revenue": 100,
"revenueScheduleType": “Divide”,
"revenueScheduleInstallmentPeriod": “Quarterly”,
"revenueScheduleInstallmentsNumber": 10,
"revenueScheduleStartDate": "2018-09-15"
}
```

Establish a quantity schedule only for an opportunity product.

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/OpportunityLineItem/00kR0000001WJJAIA4/OpportunityLineItemSchedules
 -H "Authorization: Bearer token"
```

JSON Request body

```
{
"type": “Quantity”,
"quantity": 10,
"quantityScheduleType": "Repeat",
"quantityScheduleInstallmentPeriod": “Daily”,
"quantityScheduleInstallmentsNumber": 150,
"quantityScheduleStartDate": "2020-09-15",
}
```

Delete Product Schedules

Delete all installments in a revenue or quantity schedule for opportunity products. Deleting all schedules also fires delete triggers. This resource is available in REST API version 43.0 and later.

Syntax

URI

/services/data/v***XX.X***/sobjects/OpportunityLineItem/***OpportunityLineItemId***/OpportunityLineItemSchedules

Formats

JSON, XML

HTTP methods
DELETE

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

Request body
`None`

Request parameters
`None`

Query

Executes the specified SOQL query.

When a SOQL query is executed, up to 2,000 records can be returned at a time in a synchronous request. However, to optimize performance, the returned batch can include fewer records than the limit or what's set in the request, based on the size and complexity of records queried. If the total number of results exceeds the limit or the requested number of results, the response contains the first batch of records, a `false` value for `done`, and a query locator. The query locator can be used with the Query More Results on page 302 resource to retrieve the next batch of records.

The response contains the total number of records returned by the QueryAll request (`totalSize`), a boolean indicating whether there are no more results (`done`), the URI of the next set of records (`nextRecordsUrl`), and an array of query result records (`records`).

Syntax

URI
`/services/data/v`***XX.X***`/query?q=`***query***

Formats
JSON, XML

HTTP Method
GET

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

Parameters

Parameter	Description
`q`	A SOQL query. To create a valid URI, replace spaces in the query string with a plus sign `+` or with `%20`. For example: `SELECT+Name+FROM+MyObject`. If the SOQL query string is invalid, a MALFORMED_QUERY response is returned.

Example

Example Response Body

```
{
  "totalSize": 3222,
```

```
  "done": false,
  "nextRecordsUrl": "/services/data/v60.0/query/01gRO0000016PIAYA2-500",
  "records": [
    {
      "attributes": {
        "type": "Contact",
        "url": "/services/data/v60.0/sobjects/Contact/003RO0000035WQgYAM"
      },
      "Id": "003RO0000035WQgYAM",
      "Name": "John Smith"
    },
    ...
  ]
}
```


Resources for Executing SOQL Queries

- For an example, see Execute a SOQL Query.
- To change the batch size, see Query Options Header.
- To get feedback on a query and a report using the `explain` parameter, see Get Feedback on Query Performance
- For more information on SOQL in general, see the SOQL and SOSL Reference.

Data Cloud Query Profile Parameters

Data Cloud Query and Unified Profile parameters allow you to leverage Salesforce REST API Query endpoint to execute SOQL queries against the Unified Profile, Data Source objects, or Data Model objects within your org. This functionality is supported using API version 51.0 or later.

For general information about using the Query REST call, see Execute a SOQL Query and Query.

Supported SOQL Parameters

The following SOQL parameters are supported for use with Data Cloud:

- `SELECT` statement on a single object
- `SELECT` clause: count()
- `SOQL WHERE` clause: contains operators
- `SOQL LIKE`
- `SOQL LIMIT` clause

  The default limit is set to 100. The max limit is 2,000 records in a single call.
- `SOQL OFFSET` clause
- `SOQL ORDER BY` clause

SOQL Limitations

The following queries are not supported for use with Data Cloud:

- `SOQL` Subqueries

- `SELECT` clause: aggregate functions
- `SELECT` clause: date functions
- `SOQL HAVING` clause

Sample Queries

Use Case	Queries
Data Preview: Examine data that has been ingested into a data lake object.	**Get Email Click Events** `SELECT` SubscriberKey__c, EngagementChannel__c, EmailName__c, SubjectLine__c `FROM` sfmc_email_engagement_click_{EID}__dll `LIMIT` =100
Consent Lookup: Retrieve Individual IDs from Contact Point Data Model objects based on email address, phone number, or first and last name.	**Get Individual IDs by Email Address** `SELECT` PartyId__c `FROM` ContactPointEmail__dlm `WHERE` EmailAddress__c='jjones@email.com' `LIMIT` =100 **Get Individual IDs by Phone Number** `SELECT` PartyId__c `FROM` ContactPointPhone__dlm `WHERE` TelephoneNumber__c='555-123-4567' `LIMIT` =100 **Get Individual IDs by Name** `SELECT` IndividualId__c `FROM` Individual__dlm `WHERE` FirstName__c='Jimmy' AND LastName__c='Smith' `LIMIT` =100
Unified Profile Lookup: Retrieve Unified Individual and Unified Contact Points by Source Record Id.	**Step 1:** **Get Unified Record Id by Source Record Id** `SELECT` UnifiedRecordId__c `FROM` IndividualIdentityLink__dlm `WHERE` SourceRecordID__c='{sourceID}' `LIMIT` =100 **Step 2:** **Query Unified Individual by Unified Profile ID** `SELECT` FirstName__c, LastName__c `FROM` UnifiedIndividual__dlm `WHERE` Id__c='{UnifiedRecordId__c}' `LIMIT` =100 **Step 3:** **Query Unified Contact Point Details by Unified Profile ID** *Unified Contact Point Email* `SELECT` EmailAddress__c `FROM` UnifiedContactPointEmail__dlm `WHERE` PartyId__c={UnifiedRecordId__c} `LIMIT` =100 *Unified Contact Point Phone* `SELECT` TelephoneNumber__c `FROM` UnifiedContactPointPhone__dlm `WHERE` PartyId__c={UnifiedRecordId__c} `LIMIT` =100

Query More Results

Returns the next batch of results from a SOQL query using a query locator.

If the number of results returned from a SOQL query exceeds the number of requested records or the limit, the response contains a batch of results, a `false` value for `done`, and a query locator. Use the query locator in another request to retrieve the next batch of records. If there are still more records to be returned, the response contains a new query locator and `done` is `false`. You can continue retrieving results from the initial query until `done` is `true`, which indicates that all results are returned.

The response contains the total number of records returned by the QueryAll request (`totalSize`), a boolean indicating whether there are no more results (`done`), the URI of the next set of records (`nextRecordsUrl`), and an array of query result records (`records`).

Syntax

URI

/services/data/v***XX.X***/query/***queryLocator***

Formats

JSON, XML

HTTP Method

GET

Authentication

Authorization: Bearer ***token***

Parameters

Parameter	Description
`queryLocator`	A string used to retrieve the next set of query results. If there are more results to be retrieved, the previous set of query results contains the query locator in the `nextRecordsUrl` field.

Example

Example Response Body

```
{
  "totalSize": 3222,
  "done": false,
  "nextRecordsUrl": "/services/data/v60.0/query/01gRO0000016PIAYA2-500",
  "records": [
    {
      "attributes": {
        "type": "Contact",
        "url": "/services/data/v60.0/sobjects/Contact/003RO0000035WQgYAM"
      },
      "Id": "003RO0000035WQgYAM",
      "Name": "John Smith"
    },
    ...
  ]
}
```

Resources for Executing SOQL Queries

- For an example of how to use the query locator see Execute a SOQL Query.
- For another option to change the batch size, see Query Options Header.
- For more information on SOQL in general, see the SOQL and SOSL Reference.

QueryAll

Executes the specified SOQL query. Unlike the Query resource, QueryAll returns records that are deleted because of a merge or delete. QueryAll also returns information about archived task and event records. This resource is available in REST API version 29.0 and later.

When a QueryAll request is executed, up to 2,000 records can be returned at a time in a synchronous request. However, to optimize performance, the returned batch can include fewer records than the limit or what's set in the request, based on the size and complexity of records queried. If the total number of results exceeds the limit or the requested number of results, the response contains a batch of results, a `false` value for `done`, and a query locator. The query locator can be used with the QueryAll More Results resource to retrieve the next batch of records.

Although the `nextRecordsUrl` has `query` in the URL, it still provides remaining results from the initial QueryAll request. The remaining results include deleted records that matched the initial query.

The response contains the total number of records returned by the QueryAll request (`totalSize`), a boolean indicating whether there are no more results (`done`), the URI of the next set of records (`nextRecordsUrl`), and an array of query result records (`records`).

Syntax

URI

`/services/data/vXX.X/queryAll?q=query`

Formats

JSON, XML

HTTP Method

GET

Authentication

`Authorization: Bearer token`

Parameters

Parameter	Description
`q`	A SOQL query. To create a valid URI, replace spaces with a plus sign `+` in the query string. For example: `SELECT+Name+FROM+MyObject`.

Example

Example Response Body


```
{
  "totalSize": 3222,
  "done": false,
  "nextRecordsUrl": "/services/data/v60.0/query/01gRO0000016PIAYA2-500",
  "records": [
    {
      "attributes": {
        "type": "Contact",
        "url": "/services/data/v60.0/sobjects/Contact/003RO0000035WQgYAM"
      },
      "Id": "003RO0000035WQgYAM",
      "Name": "John Smith"
```

```
        },
        ...
    ]
}
```

Resources for Executing SOQL Queries

- To run a query that includes deleted items, see Execute a SOQL Query that Includes Deleted Items.
- To increase the batch size of query results, use the query identifier, described in Execute a SOQL Query or use the Query Options Header.
- For more information about SOQL generally, see the SOQL and SOSL Reference.

QueryAll More Results

Returns the next batch of results from a QueryAll request using a query locator. This API resource executes the specified QueryAll request. This resource is available in REST API version 29.0 and later.

If the number of results returned from a SOQL query exceeds the number of requested records or the limit, the response contains a batch of results, a `false` value for `done`, and a query locator. Use the query locator in a QueryAll More Results request to retrieve the next batch of records. If there are still more records to be returned, the response contains a new query locator and `done` is `false`. You can continue retrieving results from the initial QueryAll request until `done` is `true`, which indicates that all results are returned.

Note: The URI specified in the `nextRecordsUrl` field of a QueryAll response body contains `query` instead of `queryAll`. To retrieve the next set of results, you can use either the Query More Results or the QueryAll More Results resources with the same query locator. The remaining results include deleted records that match the initial query.

For example, if the response body of a QueryAll request contains `"nextRecordsUrl": "/services/data/v60.0/query/01g5e00001AH2dOAAT-4000"`, you can retrieve the next set of QueryAll results with either URI.

- `/services/data/v60.0/query/01g5e00001AH2dOAAT-4000`
- `/services/data/v60.0/queryAll/01g5e00001AH2dOAAT-4000`

The response contains the total number of records returned by the QueryAll request (`totalSize`), a boolean indicating whether there are no more results (`done`), the URI of the next set of records (`nextRecordsUrl`), and an array of query result records (`records`).

Syntax

URI

`/services/data/vXX.X/queryAll/queryLocator`

Formats

JSON, XML

HTTP Method

GET

Authentication

`Authorization: Bearer token`

Parameters

Parameter	Description
`queryLocator`	A string used to retrieve the next set of query results. If there are more results to be retrieved, the previous set of QueryAll results contains the query locator in the `nextRecordsUrl` field.

Example

Example Response Body

```
{
  "totalSize": 3222,
  "done": false,
  "nextRecordsUrl": "/services/data/v60.0/query/01gRO0000016PIAYA2-500",
  "records": [
    {
      "attributes": {
        "type": "Contact",
        "url": "/services/data/v60.0/sobjects/Contact/003RO0000035WQgYAM"
      },
      "Id": "003RO0000035WQgYAM",
      "Name": "John Smith"
    },
    ...
  ]
}
```

Resources for Executing SOQL Queries

- To send a QueryAll request that includes deleted items, see Execute a SOQL Query that Includes Deleted Items.
- To increase the batch size of query results use the Query Options Header.
- For more information about SOQL generally, see the SOQL and SOSL Reference.

Query Performance Feedback (Beta)

Analyzes the performance of a specified SOQL query, report, or list view without executing it.

Use the `explain` parameter in a request to get a response that details how Salesforce processes your query, report, or list view and how to optimize it.

The Query Performance Feedback resource is available in API version 30.0 and later.

Note: This feature is a Beta Service. Customer may opt to try such Beta Service in its sole discretion. Any use of the Beta Service is subject to the applicable Beta Services Terms provided at Agreements and Terms.

Syntax

URI

`/services/data/vXX.X/query?explain=query`

Formats

JSON, XML

HTTP Method

GET

Authentication

`Authorization: Bearer token`

Parameters

Parameter	Description
`explain`	A SOQL query, report, or list view that you want to analyze. To analyze a query, provide the full query in the request. To analyze a report or list view, provide the ID of the report or list view. If the SOQL query string is invalid, a MALFORMED_QUERY response is returned. If the report or list view ID is invalid, an INVALID_ID response is returned.

Response body

The response body contains one or more plans that can be used to execute the query, report, or list view. The plans are sorted from most optimal to least optimal. Each plan has the following information:

Name	Type	Description
`cardinality`	number	The estimated number of records the query would return, based on index fields, if any.
`fields`	string[]	If the leading operation type is `Index`, these values are the index fields used for the query. Otherwise, the value is null.
`leadingOperationType`	string	The primary operation type that can use to optimize the query. This type can be one of these values: • Index—The query uses an index on the query object. • Other—The query uses optimizations internal to Salesforce. • Sharing—The query uses an index based on the user's sharing rules. If there are sharing rules that limit which records are visible to the current user, those rules can optimize the query. • TableScan—The query scans all records for the query object, and doesn't use an index.
`notes`	feedback note[]	An array of one or more feedback notes. Each note contains: • `description`— A detailed description of a part of the optimization. This description can include information on optimizations that can't be used and why they can't used. • `fields`— An array of one or more fields used for the optimization.

Name	Type	Description
		• `tableEnumOrId`— The table name for the fields used for the optimization. This response field is available in API version 33.0 and later.
`relativeCost`	number	The cost of this query compared to the SOQL selective query threshold. A value greater than 1.0 means the query isn't selective. For more information on selective queries, see Working with Very Large SOQL Queries in the *Apex Developer Guide*.
`sobjectCardinality`	number	The approximate count of all records in your organization for the query object.
`sobjectType`	string	The name of the query object, such as `Merchandise__c`.

Resources for Executing SOQL Queries

- For an example of how to use the `explain` parameter, see Get Feedback on Query Performance.
- For more information on SOQL in general, see the SOQL and SOSL Reference.

Quick Actions

Access global quick actions and object-specific quick actions. By using the POST method with this resource, you can create records using a quick action. This resource is available in REST API version 28.0 and later.

When working with actions, also refer to sObject Quick Actions.

IN THIS SECTION:

Get Quick Actions
Gets a list of quick actions. This resource is available in REST API version 28.0 and later.

Create Records Using Quick Actions
Creates a record via a quick action. This resource is available in REST API version 28.0 and later.

Return Headers of Quick Actions
Returns only the headers that are returned by sending a GET request to the Quick Actions resource. This gives you a chance to see the header values before retrieving the content of the resource. This resource is available in REST API version 28.0 and later.

Get Quick Actions

Gets a list of quick actions. This resource is available in REST API version 28.0 and later.

Add all required fields to an object before you create a quick action for that object. If you add a required field after creating a quick action, the field doesn't appear in the quick action's describe results. Then, when the quick action runs, the field isn't available and an error occurs for the missing field. If you don't want the required field to appear in the quick action's layout, set a default value for the field.

When working with actions, also refer to sObject Quick Actions.

Syntax

URI

`/services/data/vXX.X/quickActions/`

Formats

JSON, XML

HTTP Method

GET

Authentication

`Authorization: Bearer token`

Parameters

None required

Example

Example Request

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/quickActions/ -H
"Authorization: Bearer token"
```

Create Records Using Quick Actions

Creates a record via a quick action. This resource is available in REST API version 28.0 and later.

Add all required fields to an object before you create a quick action for that object. If you add a required field after creating a quick action, the field doesn't appear in the quick action's describe results. Then, when the quick action runs, the field isn't available and an error occurs for the missing field. If you don't want the required field to appear in the quick action's layout, set a default value for the field.

When working with actions, also refer to sObject Quick Actions.

Syntax

URI

`/services/data/vXX.X/quickActions/`

Formats

JSON, XML

HTTP Method

POST

Authentication

`Authorization: Bearer token`

Parameters

None required

Example

Example Request

```
curl -X POST
https://MyDomainName.my.salesforce.com/services/data/v60.0/quickActions/CreateContact
-H "Authorization: Bearer token" -H "Content-Type: application/json" -d
@exampleRequestBody.json
```

Example Request Body

```
{

   "record" : { "LastName" : "Smith" }

}
```

Return Headers of Quick Actions

Returns only the headers that are returned by sending a GET request to the Quick Actions resource. This gives you a chance to see the header values before retrieving the content of the resource. This resource is available in REST API version 28.0 and later.

Add all required fields to an object before you create a quick action for that object. If you add a required field after creating a quick action, the field doesn't appear in the quick action's describe results. Then, when the quick action runs, the field isn't available and an error occurs for the missing field. If you don't want the required field to appear in the quick action's layout, set a default value for the field.

When working with actions, also refer to sObject Quick Actions.

URI
`/services/data/vXX.X/quickActions/`

Formats
JSON, XML

HTTP Method
HEAD

Authentication
`Authorization: Bearer token`

Parameters
None required

Example

Example Request

```
curl -X HEAD --head
https://MyDomainName.my.salesforce.com/services/data/v60.0/quickActions/ -H
"Authorization: Bearer token"
```

Recent List Views

Returns the list of recently used list views for the given sObject type. This resource is available in REST API version 32.0 and later.

Syntax

URI

`/services/data/vXX.X/sobjects/sObject/listviews/recent`

Formats

JSON, XML

HTTP Method

GET

Authentication

`Authorization: Bearer token`

Parameters

None

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/Account/listviews/recent
 -H "Authorization: Bearer token"
```

Example Response Body


```
{
  "done" : true,
  "listviews" : [ {
    "describeUrl" :
"/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcCNMA0/describe",
    "developerName" : "MyAccounts",
    "id" : "00BD0000005WcCNMA0",
    "label" : "My Accounts",
    "resultsUrl" :
"/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcCNMA0/results",
    "soqlCompatible" : true,
    "url" : "/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcCNMA0"
  }, {
    "describeUrl" :
"/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcBeMAK/describe",
    "developerName" : "NewThisWeek",
    "id" : "00BD0000005WcBeMAK",
    "label" : "New This Week",
    "resultsUrl" :
"/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcBeMAK/results",
    "soqlCompatible" : true,
    "url" : "/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcBeMAK"
  }, {
    "describeUrl" :
"/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcCFMA0/describe",
    "developerName" : "AllAccounts",
    "id" : "00BD0000005WcCFMA0",
    "label" : "All Accounts",
    "resultsUrl" :
```

```
"/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcCFMA0/results",
    "soqlCompatible" : true,
    "url" : "/services/data/v60.0/sobjects/Account/listviews/00BD0000005WcCFMA0"
  } ],
  "nextRecordsUrl" : null,
  "size" : 3,
  "sobjectType" : "Account"
}
```


Recently Viewed Items

Gets the most recently accessed items that were viewed or referenced by the current user. Salesforce stores information about record views in the interface and uses it to generate a list of recently viewed and referenced records, such as in the sidebar and for the auto-complete options in search.

This resource only accesses most recently used item information. If you want to modify the list of recently viewed items, you'll need to update recently viewed information directly by using a SOQL Query with a `FOR VIEW` or `FOR REFERENCE` clause.

URI

`/services/data/vXX.X/recent`

Formats

JSON, XML

HTTP Method

GET

Authentication

`Authorization: Bearer token`

Parameters

Parameter	Description
`limit`	An optional limit that specifies the maximum number of records to be returned. If this parameter is not specified, the default maximum number of records returned is the maximum number of entries in RecentlyViewed, which is 200 records per object.

Example

- For an example of retrieving a list of recently viewed items, see View Recently Viewed Records on page 80.
- For an example of setting records as recently viewed, see Mark Records as Recently Viewed on page 81.

Record Count

Lists information about object record counts in your organization.

This resource is available in REST API version 40.0 and later for API users with the "View Setup and Configuration" permission. The returned record count is approximate, and does not include the following types of records:

- Deleted records in the recycle bin.
- Archived records.

Syntax

URI

/services/data/v***XX.X***/limits/recordCount?sObjects=***objectList***

Formats

JSON, XML

HTTP Method

GET

Authentication

Authorization: Bearer ***token***

Parameters

Parameter	Description
sObjects	A comma-delimited list of object names. If a listed object is not found in the org, it is ignored and not returned in the response. This parameter is optional. If this parameter is not provided, the resource returns record counts for all objects in the org.

Response body

Record Count Response Body

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/limits/recordCount?sObjects=Account,Contact
 -H "Authorization: Bearer token"
```

Example Response Body

```
{
   "sObjects" : [ {
     "count" : 3,
     "name" : "Account"
   }, {
     "count" : 10,
     "name" : "Contact"
   } ]
}
```

Record Count Response Body

Describes the result of a Record Count request.

Record Count Results

Properties

Name	Type	Description
`sObjects`	Record Count sObject Result[]	Collection of sObject record count results. The order of objects in the collection is not guaranteed to match the order of objects in the request.

JSON example

```
{
   "sObjects" : [ {
     "count" : 3,
     "name" : "Account"
   }, {
     "count" : 10,
     "name" : "Contact"
   } ]
}
```

Record Count sObject Result

Properties

Name	Type	Description
`count`	Integer	The number of records for the object in the org. This is an approximate count and does not include soft-deleted or archived records.
`name`	String	The name of the object.

JSON example

```
{
   "count" : 10,
   "name" : "Contact"
}
```

sObject Relevant Items

Gets the current user's most relevant items. Relevant items include records for objects in the user's global search scope and also most recently used (MRU) objects.

Relevant items include up to 50 of the most recently viewed or updated records for each object in the user's global search scope.

Note: The user's global search scope includes the objects the user interacted with most in the last 30 days, including objects the user pinned from the search results page in the Salesforce Classic.

Then, the resource finds more recent records for each most recently used (MRU) object until the maximum number of records, which is 2,000, is returned.

This resource only accesses the relevant item information. Modifying the list of relevant items is not currently supported.

This resource is available in API version 35.0 and later.

Syntax

URI

`/services/data/vXX.X/sobjects/relevantItems`

Formats

JSON

HTTP Method

GET

Authentication

`Authorization: Bearer token`

Parameters

Parameter	Description
`lastUpdatedId`	Optional. Compares the entire current list of relevant items to a previous version, if available. Specify the `lastUpdatedId` value returned in a previous response.
`sobjects`	Optional. To scope the results to a particular object or set of objects, specify the name for one or more sObjects. Note: sObject names are case-sensitive.
`sobject.lastUpdatedId`	Optional. Compares the current list of relevant items for this particular object to a previous version, if available. Specify the `lastUpdatedId` value returned in a previous response. Note: You can only specify this parameter for the sObjects specified in the `sobjects` parameter.

Response header

The response contains headers unique to this resource.

Name	Type	Description
`lastUpdatedId`	string	A unique code that can be used in subsequent calls to compare the results for a complete result set with the results in this response list.
`newResultSetSinceLastQuery`	boolean (`true` or `false`)	If a response was previously requested for the current user, indicates whether the current response matches the previous response, or the one specified by a `lastUpdatedId`.

Response body

The response contains an array of records for each object returned, including the following information for each record.

Name	Type	Description
`apiName`	string	The object's unique name, such as `Account`
`key`	ID	The first 3 characters of the sObject's ID that indicates the object type.
`label`	string	The object's plural label, such as `Accounts`.
`lastUpdatedId`	string	A unique code that can be used in subsequent calls to compare the results for the new result set with the current results for this object.
`qualifiedApiName`	string	A unique external name for the sObject.
`recordIds`	ID	A comma-separated list of IDs for the matching records.

Example

See View Relevant Items.

Get Knowledge Language Settings

Gets the existing Knowledge language settings, including the default knowledge language and a list of supported Knowledge language information. This resource can be used in API version 31.0 and later.

Salesforce Knowledge must be enabled in your organization. It gets the Knowledge language settings, including the default knowledge language and a list of supported Knowledge language information.

Syntax

URI

`/services/data/v`***`XX.X`***`/knowledgeManagement/settings`

Formats

JSON, XML

HTTP methods

GET

Authentication

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

Request body

None required

Request parameters

None

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/knowledgeManagement/settings
 -H "Authorization: Bearer token"
```

Example Response Body


```
{
    "defaultLanguage" : "en_US",
    "knowledgeEnabled" : true,
    "languages" : [ {
    "active" : true,
    "name" : "en_US"
    }, {
    "active" : true,
    "name" : "it"
    }, {
    "active" : true,
    "name" : "zh_CN"
    }, {
    "active" : true,
    "name" : "fr"
    } ]
}
```


Search

Executes the specified SOSL search. The search string must be URL-encoded.

For more information on SOSL see the *SOQL and SOSL Reference*.

Syntax

URI

`/services/data/vXX.X/search/?q=SOSL_searchString`

Formats

JSON, XML

HTTP Method

GET

Authentication

`Authorization: Bearer token`

Parameters

Parameter	Description
`q`	A SOSL statement that is properly URL-encoded.

Example

See Search for a String on page 64.

Search Scope and Order

Returns an ordered list of objects in the default global search scope of a logged-in user. Global search keeps track of which objects the user interacts with and how often, and arranges the search results accordingly. Objects used most frequently appear at the top of the list.

The returned list reflects the object order in the user's default search scope, including any pinned objects on the user's search results page. This call is useful if you want to implement a custom search results page using the optimized global search scope. The search string must be URL-encoded.

Syntax

URI

`/services/data/vXX.X/search/scopeOrder`

Formats

JSON, XML

HTTP Method

GET

Authentication

`Authorization: Bearer token`

Example

See Get the Default Search Scope and Order.

Search Result Layouts

Returns search result layout information for the objects in the query string. For each object, this call returns the list of fields displayed on the search results page as columns, the number of rows displayed on the first page, and the label used on the search results page.

This call supports bulk fetch for up to 100 objects in a query.

Syntax

URI

`/services/data/vXX.X/search/layout/?q=commaDelimitedObjectList`

Formats

JSON, XML

HTTP Method

GET

Authentication

`Authorization: Bearer token`

Response format

Property	Type	Description
`field`	String	Object and field name formatted with a period separating. For example: `Account.Name`.
`format`	String	The type of date field, such as the date only or date and time. Only date related types are specified; otherwise, `null`.
`label`	String	Name as it appears to users
`name`	String	API name

Example

See Get Search Result Layouts for Objects.

Lightning Toggle Metrics

Returns details about users who switched between Salesforce Classic and Lightning Experience. This resource is available in REST API version 44.0 and later.

Use this object with the following APIs:

- Platform
- Metadata API
- Tooling API

Syntax

URI

`/services/data/vXX.X/sobjects/LightningToggleMetrics`

Formats

JSON, XML

HTTP methods

GET

Authentication

`Authorization: Bearer tokens`

Request parameters

Parameter	Description
`UserId`	The user ID.
`RecordCount`	The count of records returned.

Parameter	Description
`MetricsDate`	The date the switch was recorded.
`Action`	Did the user switch to Salesforce Classic or Lightning Experience.

Example

```
SELECT sum(RecordCount) Total FROM LightningToggleMetrics WHERE MetricsDate = LAST_MONTH
AND Action = 'switchToAloha'
```

Lightning Usage by App Type

Returns the total number of Lightning Experience and Salesforce Mobile users. This resource is available in REST API version 44.0 and later.

Use this object with the following APIs:

- Platform
- Metadata API
- Tooling API

Syntax

URI

`/services/data/vXX.X/sobjects/LightningUsageByAppTypeMetrics`

Formats

JSON, XML

HTTP methods

GET

Authentication

`Authorization: Bearer token`

Request parameters

Parameter	Description
`AppExperience`	The app used: • Salesforce Mobile • Lightning Experience
`MetricsDate`	The date the data was recorded.
`UserID`	The user ID.

Example

```
SELECT MetricsDate,user.profile.name,COUNT_DISTINCT(user.id) Total FROM
LightningUsageByAppTypeMetrics WHERE MetricsDate = LAST_N_DAYS:30 AND AppExperience =
'Salesforce Mobile' GROUP BY MetricsDate,user.profile.name
```

Lightning Usage by Browser

Returns Lightning Experience usage results grouped by browser instance. This resource is available in REST API version 44.0 and later.

Use this object with the following APIs.

- Platform
- Metadata API
- Tooling API

Syntax

URI

`/services/data/vXX.X/sobjects/LightningUsageByBrowserMetrics`

Formats

JSON, XML

HTTP methods

GET

Authentication

`Authorization: Bearer token`

Request body

SOQL Query.

Request parameters

Parameter	Description
`Browser`	The browser used.
`EptBin3To5`	Number of times that a page loaded between 3-5 seconds.
`EptBin5To8`	Number of times that a page loaded between 5-8 seconds.
`EptBin8To10`	Number of times that a page loaded between 8-10 seconds.
`EptBinOver10`	Number of times that a page loaded over 10 seconds.
`EptBinUnder3`	Number of times that a page loaded under 3 seconds.
`MetricsDate`	The date the metric was recorded.
`PageName`	The name of the page.
`RecordCountEPT`	Number of records for a page/browser where the valid EPT was recorded.
`SumEPT`	Sum of the EPT values for page/browser.

Parameter	Description
`TotalCount`	Total records for a page/browser.

Example

Example Request Body

```
SELECT CALENDAR_MONTH(MetricsDate) MetricsDate, Browser Browser, SUM(TotalCount) Total
 FROM LightningUsageByBrowserMetrics WHERE MetricsDate = Last_N_Months:3 AND (NOT Browser
 like 'OTHER%') GROUP BY CALENDAR_MONTH(MetricsDate),Browser
```

Lightning Usage by Page

Represents standard pages users viewed most frequently in Lightning Experience. This resource is available in REST API version 44.0 and later.

Use this object with the following APIs:

- Platform
- Metadata API
- Tooling API

Syntax

URI

`/services/data/vXX.X/sobjects/LightningUsageByPageMetrics`

Formats

JSON, XML

HTTP methods

GET

Authentication

`Authorization: Bearer token`

Parameter	Description
`EptBin3To5`	Number of times that a page loaded between 3-5 seconds.
`EptBin5To8`	Number of times that a page loaded between 5-8 seconds.
`EptBin8To10`	Number of times that a page loaded between 8-10 seconds.
`EptBinOver10`	Number of times that a page loaded over 10 seconds.
`EptBinUnder3`	Number of times that a page loaded under 3 seconds.
`PageName`	The name of the page.
`MetricsDate`	The date the metric was recorded.

Parameter	Description
`RecordCountEPT`	Number of records for a page/user where the valid EPT was recorded.
`SumEPT`	Sum of the EPT values for a page/user.
`TotalCount`	Total records for a page/user.
`UserId`	User ID.

Example

```
SELECT TotalCount FROM LightningUsageByPageMetrics ORDER BY PageName ASC NULLS FIRST LIMIT
 10
```

Lightning Usage by FlexiPage

Returns details about the custom pages viewed most frequently in Lightning Experience. This resource is available in REST API version 44.0 and later.

Use this object with the following APIs:

- Platform
- Metadata API
- Tooling API

Syntax

URI

`/services/data/v`***`XX.X`***`/sobjects/LightningUsageByFlexiPageMetrics`

Formats

JSON, XML

HTTP methods

GET

Authentication

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

Request parameters

Parameter	Description
`FlexiPageNameOrId`	Namespace and file name, or Page ID of `FlexiPage` files.
`FlexiPageType`	The `FlexiPage` type. For example, record details are displayed using `RecordPage`" type.
`MetricsDate`	The date the metric was recorded.
`RecordCountEPT`	Number of records for a `FlexiPage` type, where the valid EPT was recorded.

Parameter	Description
SumEPT	Sum of the EPT values for a record
TotalCount	Total records for a type.

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/sobjects/LightningUsageByFlexiPageMetrics
 -H "Authorization: Bearer token"
```

Example Request Body

```
SELECT FlexiPageNameOrId FlexiPageNameOrId, SUM(TotalCount) Total FROM
LightningUsageByFlexiPageMetrics WHERE MetricsDate = Last_N_DAYS:7 AND (NOT
FlexiPageNameOrId = 'unknown unknown') AND (NOT FlexiPageNameOrId = 'unknown | unknown')
 GROUP BY FlexiPageNameOrId ORDER BY SUM(TotalCount) Desc Limit 10
```

Lightning Exit by Page Metrics

Returns frequency metrics about the standard pages within which users switched from Lightning Experience to Salesforce Classic. This resource is available in REST API version 44.0 and later.

Use this object with the following APIs:

- Platform
- Metadata API
- Tooling API

Syntax

URI

/services/data/v***XX.X***/sobjects/LightningExitByPageMetrics

Formats

JSON, XML

HTTP methods

GET

Authentication

Authorization: Bearer *token*

Request parameters

Parameter	Description
MetricsDate	The date the data was recorded.
PageName	Current Page from which User Switched from Lightning to Aloha

Parameter	Description
`RecordCount`	The number of records per user and page.
`UserId`	The user ID.

Example

```
SELECT PageName PageName, SUM(RecordCount) Total FROM LightningExitByPageMetrics WHERE
MetricsDate = Last_N_DAYS:7 GROUP BY PageName ORDER BY SUM(RecordCount) Desc Limit 10
```

Salesforce Scheduler Resources

Use Salesforce Scheduler REST APIs to get appointment time slots or available service resources based on work type groups and service territories.

IN THIS SECTION:

SEE ALSO:

Connect REST API Developer Guide: Lightning Scheduler Resources

Scheduling

Returns a list of available Salesforce Scheduler REST resources and corresponding URIs. This resource is available in REST API version 45.0 and later.

Syntax

URI

`/services/data/vXX.X/scheduling/`

Formats

JSON, XML

HTTP methods

GET

Authentication

`Authorization: Bearer token`

Example

Example Response Body

```
{
  "getAppointmentCandidates" : "/services/data/v60.0/scheduling/getAppointmentCandidates",

   "getAppointmentSlots" : "/services/data/v60.0/scheduling/getAppointmentSlots"
}
```

Get Appointment Slots

Returns a list of available appointment time slots for a resource based on given work type group or work type and service territories.

The appointment time slots are determined based on your Salesforce Scheduler data model configurations. Here are some prerequisites that you can consider while setting up data.

- Set up Salesforce Scheduler before making your requests. The setup includes creating or configuring Service Resources, Service Territory Members, Work Type Groups, Work Types, Work Type Group Members, and Service Territory Work Types. See Manage Business Information in Salesforce Scheduler for more information.
- Configure a work type mapped for each territory in the request body via Service Territory Work Type. Map the same work type to the work type group, via work type group member.

The following factors affect how time slots are calculated and returned.

- Timezones that differ across operating hours are handled and results are always returned in UTC.
- The resource must be marked as a required resource on the assigned resource object.
- The resource is considered unavailable If the status categories of the resource assigned to service appointments are other than `Canceled`, `Cannot Complete`, and `Completed`.
- Resource Absences of all types are considered unavailable from start to end.
- The following fields of Work Type records, if configured, are used to fine-tune time slot requirements. For more information, see Create Work Types in Salesforce Scheduler.

Parameter	Description
`Timeframe Start`	Time slots sooner than `current time` + ***`Timeframe Start`*** aren't returned.
`Timeframe End`	Time slots later than `current time` + ***`Timeframe End`*** aren't returned.
`Block Time Before Appointment`	The time period before the appointment is considered as unavailable.
`Block Time After Appointment`	The time period after the appointment is considered as unavailable.
`Operating Hours`	The overlap of all operating hours from the account, work type, service territory, and service territory member are considered while determining time slots. For more information, see Set Up Operating Hours in Salesforce Scheduler.

- Only the time slots within the period of 31 days from the start date are returned.
- Salesforce Scheduler uses multiple factors, such as field values, scheduled appointments, absences, Scheduler Settings, and Scheduling Policies to determine available time slots, including the earliest and latest appointment slots. See How Does Salesforce Scheduler Determine Available Time Slots.

  Note: If asset scheduling is enabled, you can provide an asset-based service resource in `requiredResourceIds` to retrieve available timeslots for the asset resource.

Syntax

URI

`/services/data/vXX.X/scheduling/getAppointmentSlots`

Available version

45.0

Formats

JSON, XML

HTTP methods

POST

Authentication

`Authorization: Bearer token`

Request body

Parameter	Required	Type	Description
`accountId`	No	String	The ID of the associated account.
`allowConcurrentScheduling`	No	Boolean	If true, allows scheduling of concurrent appointments in a time slot. If false, concurrent appointments aren't allowed. The default is false. Available in API version 47.0 and later.
`correlationId`	No	String	The ID to pass custom information to the `ServiceResourceScheduleHandler` Apex interface. For example, you can use the correlation ID to identify the app, website, or any other external system that calls this Apex interface implementation. If you don't pass a custom value, a randomly generated identifier is passed. This field is available in API version 53.0 and later.
`endTime`	No	String	The latest time that a time slot can end (inclusive).
`engagementChannelTypeIds`	No	String[]	The ID of the engagement channel type record. The availability of time slots is filtered based on the engagement channel type selected. This field is available in API version 56.0 and later. Note: This field supports only one engagement channel type ID. You can use engagement channel types with the `getAppointmentSlots` API only if:

Parameter	Required	Type	Description
			• The **Schedule Appointments Using Engagement Channels** setting is enabled in Salesforce Scheduler Settings in your Salesforce org. • Shifts are defined in the scheduling policy. For more information on setting up shifts in scheduling policy, see Define Shift Rules in Scheduling Policy. Note: Engagement channel types are not supported with operating hours rules in the scheduling policy.
`primaryResourceId`	No	String	The ID of the primary resource in multi-resource scheduling. This field is available in API version 48.0 and later. Note: This field is required only when multi-resource scheduling is enabled.
`requiredResourceIds`	Yes	String[]	List of resource IDs that must be available during the time slot.
`schedulingPolicyId`	No	String	The ID of the AppointmentSchedulingPolicy object. If no scheduling policy is passed in the request body, the default configurations are used. The only scheduling policy configuration that is used in determining time slots is the enforcement of account visiting hours.
`startTime`	No	String	The earliest time that a time slot can begin (inclusive). Defaults to the current time of the request, if empty.
`territoryIds`	Yes	String[]	List of IDs of service territories, where the work that is being requested is performed.
`workType`	Required if `workTypeGroupId` isn't specified.	Work Type	The type of the work to be performed.
`workTypeGroupId`	Required if `workType` isn't given.	String	The ID of the work type group containing the work types that are being performed.

Note: To determine the required fields in your request body, consider the following points:

- Provide either the `workTypeGroupId` or `workType` parameter in your request body, but not both.
- If the `workType` parameter is specified, then you must provide either the `id` or `durationInMinutes` parameter.
- If `id` of the `workType` parameter is specified, then the rest of the `workType` fields are optional.

Response Body

Execution of a successful request returns the response body containing a list of available time slots.

Parameter	Required	Type	Description
`timeSlots`	Yes	Time Slots on page 335 [ ]	List of time slots included in each territory.

Example

Example Request Body

Using `workTypeGroupId`:

```
{
  "startTime": "2019-01-23T00:00:00.000Z",
  "endTime": "2019-02-28T00:00:00.000Z",
  "workTypeGroupId": "0VSB0000000KyjBOAS",
  "accountId": "001B000000qAUAWIA4",
  "territoryIds": [
    "0HhB0000000TO9WKAW"
  ],
  "schedulingPolicyId": "0VrB0000000KyjB",
  "requiredResourceIds": [
    "0HnB0000000TO8gKAK"
  ],
  "engagementChannelTypeIds": [
    "0eFRM00000000Bv2AI"
  ]
}
```

Using `workType`:

```
{
  "startTime": "2019-01-23T00:00:00.000Z",
  "endTime": "2019-02-28T00:00:00.000Z",
  "workType": {
    "id": "08qRM00000003fkYAA"
  },
  "requiredResourceIds": [
    "0HnB0000000TO8gKAK"
  ],
  "territoryIds": [
    "0HhRM00000003OZ0AY"
  ],
  "accountId": "001B000000qAUAWIA4",
  "schedulingPolicyId": "0VrB0000000KyjB",
  "engagementChannelTypeIds": [
    "0eFRM00000000Bv2AI"
  ]
}
```

Example Response Body

```
{
  "timeSlots": [
    {
```

```
      "endTime": "2019-01-21T19:15:00.000+0000",
      "startTime": "2019-01-21T16:15:00.000+0000",
      "territoryId": "0HhB0000000TO9WKAW"
    },
    {
      "endTime": "2019-01-21T19:30:00.000+0000",
      "startTime": "2019-01-21T16:30:00.000+0000",
      "territoryId": "0HhB0000000TO9WKAW"
    },
    {
      "endTime": "2019-01-21T19:45:00.000+0000",
      "startTime": "2019-01-21T16:45:00.000+0000",
      "territoryId": "0HhB0000000TO9WKAW"
    }
  ]
}
```


Get Appointment Candidates

Returns a list of service resources (appointment candidates) based on work type group or work type and service territories.

Set up Salesforce Scheduler before making requests. This setup includes creating or configuring Service Resources, Service Territory Members, Work Type Groups, Work Types, Work Type Group Members, and Service Territory Work Types. See Set Up Salesforce Scheduler for more information.

The appointment time slots are determined based on multiple factors, such as field values, scheduled appointments, absences, Scheduler Settings, and Scheduling Policies to determine available time slots. See How Salesforce Scheduler Determines Available Time Slots for more information.

The following factors are considered for returning start time and end time of resources.

Resource Availability
: Determined using service territory member, service territory, work type, and account operating hours fields.

Resource Unavailability
: Determined by resource absences, existing appointments that the resource is assigned to. The resource must be marked as a required resource for the appointment with a status that isn't in closed, canceled, or completed.

Appointment Start Time Interval in the Scheduling Policy
: Appointment start time interval field in the Scheduling Policy is used to determine when the appointment can start. This interval can be 5, 10, 15, 20, 30, or 60. By default, it's set to 15.

Work Type Duration
: The end time is calculated as start time + duration of the work type.

Note: If asset scheduling is enabled, the response also includes asset-based candidates.

Syntax

URI
: `/services/data/vXX.X/scheduling/getAppointmentCandidates`

Available version
: 45.0

Formats

JSON, XML

HTTP methods

POST

Request body

Parameter	Required	Type	Description
`accountId`	No	String	The ID of the associated account.
`allowConcurrentScheduling`	No	Boolean	If true, allows scheduling of concurrent appointments in a time slot. If false, concurrent appointments aren't allowed. The default is false. This field is available in API version 47.0 and later.
`correlationId`	No	String	The ID to pass custom information to the `ServiceResourceScheduleHandler` Apex interface. For example, you can use the correlation ID to identify the app, website, or any other external system that calls this Apex interface implementation. If you don't pass a custom value, a randomly generated identifier is passed. This field is available in API version 53.0 and later.
`endTime`	No	String	The latest time that a time slot can end (inclusive). Note: The API only returns time slots up to 31 days from the `startTime`.
`engagementChannelTypeIds`	No	String[]	The ID of the engagement channel type record. The availability of service resources is filtered based on the engagement channel type selected. This field is available in API version 56.0 and later. Note: This field supports only one engagement channel type ID. You can use engagement channel types with the `getAppointmentCandidates` API only if: • The **Schedule Appointments Using Engagement Channels** setting is enabled in Salesforce Scheduler Settings in your Salesforce org. • Shifts are defined in the scheduling policy. For more information on setting up shifts in scheduling policy, see Define Shift Rules in Scheduling Policy. Note: Engagement channel types are not supported with operating hours rules in the scheduling policy.
`filterByResources`	No	String[]	A comma-separated list of service resource IDs. API returns only eligible service resources that are both in the list and in the selected service territory. The resources are sorted by the order in which the resource IDs are passed. Available in API version 51.0 and later.

Parameter	Required	Type	Description
resourceLimitApptDistribution	No	Integer	Specify the maximum number of service resources that you want to show during appointment scheduling when appointment distribution is enabled. Available in API version 53.0 and later. Note: The filterByResources field takes precedence over the resourceLimitApptDistribution field.
startTime	No	String	The earliest time that a time slot can begin (inclusive). Defaults to the current time of the request, if empty. You can also use a time from the past.
schedulingPolicyId	No	String	The ID of the AppointmentSchedulingPolicy object. If no scheduling policy is passed in the request body, the default configurations are used. All Scheduling Policy Configurations are considered when using this API.
territoryIds	Yes	String[]	List of service territory IDs, where the work that is being requested is performed.
workType	Required if workTypeGroupId isn't given.	Work Type	The type of the work to be performed.
workTypeGroupId	Required if workType isn't given.	String	The ID of the work type group containing the work types that are being performed.

Note: To determine the required fields in your request body, consider the following points:

- Provide either the `workTypeGroupId` or `workType` parameter in your request body, but not both.
- If the `workType` parameter is specified, then you must provide either the `id` or `durationInMinutes` parameter.
- If `id` of the `workType` parameter is specified, then the rest of the `workType` fields are optional.

Response Body

Execution of a successful request returns the response body containing a list of available appointment resources.

Parameter	Required	Type	Description
candidates	Yes	Candidates on page 336 [ ]	List of available appointment candidates.

Examples

Example Request Body

Using `workTypeGroupId`:

```
{
  "startTime": "2019-01-23T00:00:00.000Z",
```

```
  "endTime": "2019-02-30T00:00:00.000Z",
  "workTypeGroupId": "0VSB0000000KyjBOAS",
  "accountId": "001B000000qAUAWIA4",
  "territoryIds": [
    "0HhB0000000TO9WKAW"
  ],
  "schedulingPolicyId": "0VrB0000000KyjB",
  "engagementChannelTypeIds": [
    "0eFRM00000000Bv2AI"
  ]
}
```

Using `workTypeId`:

```
{
  "startTime": "2019-01-23T00:00:00.000Z",
  "endTime": "2019-02-30T00:00:00.000Z",
  "workType": {
    "id": "08qRM00000003fkYAA"
  },
  "territoryIds": [
    "0HhRM00000003OZ0AY"
  ],
  "accountId": "001B000000qAUAWIA4",
  "schedulingPolicyId": "0VrB0000000KyjB",
  "engagementChannelTypeIds": [
    "0eFRM00000000Bv2AI"
  ]
}
```

Example Response Body

```
{
  "candidates": [
    {
      "endTime": "2019-01-23T19:15:00.000+0000",
      "resources": [
        "0HnB0000000D2DsKAK"
      ],
      "startTime": "2019-01-23T16:15:00.000+0000",
      "territoryId": "0HhB0000000TO9WKAW",
      "engagementChannelTypeIds": [
        "0eFRM00000000Bv2AI"
      ]
    },
    {
      "endTime": "2019-01-23T19:30:00.000+0000",
      "resources": [
        "0HnB0000000D2DsKAK"
      ],
      "startTime": "2019-01-23T16:30:00.000+0000",
      "territoryId": "0HhB0000000TO9WKAW",
      "engagementChannelTypeIds": [
        "0eFRM00000000Bv2AI"
      ]
```

```
    },
    {
      "endTime": "2019-01-23T19:45:00.000+0000",
      "resources": [
        "0HnB0000000D2DsKAK"
      ],
      "startTime": "2019-01-23T16:45:00.000+0000",
      "territoryId": "0HhB0000000TO9WKAW",
      "engagementChannelTypeIds": [
        "0eFRM00000000Bv2AI"
      ]
    }
  ]
}
```

Request Bodies

To perform a POST, PATCH, or PUT request, create a request body formatted in either XML or JSON. This chapter lists the request bodies.

IN THIS SECTION:

Work Type

Details about the type of work to be performed.

Skill Requirement

List of skills that are required to complete a particular task for a work type.

Work Type

Details about the type of work to be performed.

Name	Type	Required	Description
`id`	String	Required if `durationInMinutes` is not given.	Id of the work type.
`durationInMinutes`	Integer	Required if `id` is not given.	Contains the event length, in minutes.
`timeframeStartInMinutes`	String	No	The beginning of the timeframe.
`timeframeEndInMinutes`	String	No	The end of the timeframe.
`blockTimeBeforeAppointmentInMinutes`	String	No	The time period before the appointment is considered as unavailable.
`blockTimeAfterAppointmentInMinutes`	String	No	The time period after the appointment is considered as unavailable.
`operatingHoursId`	String	No	The overlap of all operating hours from the account, work type, service territory, and service territory member are considered while determining time slots.

Name	Type	Required	Description
`skillRequirements`	Skill Requirement[]	No	List of skills that are required to complete a particular task for a work type.

Note: Provide either `Id` or `durationInMinutes` in the request body, but not both.

Skill Requirement

List of skills that are required to complete a particular task for a work type.

Name	Type	Required	Description
`skillId`	String	Yes	The skill that is required.
`SkillLevel`	String	No	The level of the skill required. Skill levels can range from zero to 99.99. Depending on your business needs, you might want the skill level to reflect years of experience, certification levels, or license classes.

Response Bodies

Successful execution of a request to a Salesforce Scheduler resource can return a response body either in JSON or XML format. For example, the request to get appointment time slots returns a list of available time slots for a selection of work type group and territories.

IN THIS SECTION:

Time Slots

Describes the result of Get Appointments Slots request.

Candidates

Describes the result of Get Appointments Candidates request.

Time Slots

Describes the result of Get Appointments Slots request.

List of time slots available for each territory.

Name	Type	Description
`endTime`	String	The end time of the appointment time slot.
`engagementChanneltypeIds`	String[]	The engagement channel type ID associated with this time slot. This field is available in API version 56.0 and later.
`remainingAppointments`	Integer	The number of appointments available in the time slot. `Appointments available in a time slot = Maximum number of appointments defined for the`

Name	Type	Description
		`time slot - Number of appointments scheduled so far in the time slot`
`startTime`	String	The start time of the appointment time slot.
`territoryId`	String	The service territory associated with this time slot.

Candidates

Describes the result of Get Appointments Candidates request.

List of available service resources.

Name	Type	Description
`endTime`	String	The end time of the appointment time slot.
`engagementChanneltypeIds`	String[]	The engagement channel type ID associated with this resource for that time slot. This field is available in API version 56.0 and later.
`resources`	String[]	List of service resource IDs that are available. Important: At present, only one resource is returned on this list. If there is more than one resource included in a territory, a new child object is added for each resource in the response JSON body.
`startTime`	String	The start time of the appointment time slot.
`territoryId`	String	The service territory associated with this resource.

Search for Records Suggested by Autocomplete and Instant Results

Returns a list of suggested records whose names match the user's search string. The suggestions resource provides autocomplete results and instant results for users to navigate directly to likely relevant records, before performing a full search. This resource is available in REST API version 32.0 and later.

The suggestions resource returns records when the record's name field includes the exact text in the search string. The last term in the search string can match the beginning of a word. Records that contain the search string within a word aren't considered a match.

Note: If the user's search query contains quotation marks or wildcards, those symbols are automatically removed from the query string in the URI.

For example, the text string *`national u`* is treated as *`national u*`* and returns "National Utility", "National Urban Company", and "First National University".

The suggestions resource returns display-ready data about likely relevant records that the user can access. A relevance algorithm determines the order of results. Each suggested record in the results contains these elements:

Element	Description
`Attributes`	The record's object type and the URL for accessing the record. Also includes the requested lookup fields' values. For example, if you requested `fields=Id,Name`, the result would include the ID and name.
`Name` (or `Title`)	The record's Name field. In the absence of a standard Name field, the Title field is used for these objects: • Dashboard • Idea • IdeaTheme • Note • Question In the absence of a standard Name or Title field, the main identifying field is used. For example, in cases, the Case Number is used.
`Id`	The record's unique identifier.

The suggestions resource supports all searchable objects except the following.

- ContentNote
- Event
- External objects
- FeedComment
- FeedPost
- IdeaComment
- Pricebook2
- Reply
- TagDefinition
- Task

Syntax

URI

`/services/data/vXX.X/search/suggestions?q=searchString&sobject=objectTypes`

Formats

JSON, XML

HTTP methods

GET

Authentication

`Authorization: Bearer token`

Request body

None required

Request parameters

Parameter	Description
fields	Optional. Used for creating lookup queries. Specify multiple fields using a comma-separated list. Specifies which lookup fields to be returned in the response.
dynamicFields	Optional. Available in API version 48.0 and later. Used to return additional dynamic fields. Specify multiple options using a comma-separated list. For example, if `dynamicFields=secondaryField` then each suggested record in the results contains an additional field besides `Id` and `Name` (or `Title`) based on the next eligible field in the search layout.
groupId	Optional. Specifies one or more unique identifiers of one or more groups that the question to return was posted to. Specify multiple groups using a comma-separated list. This parameter is only applicable when the parameter type equals `question`. Don't use with the `userId`.
ignoreUnsupportedSObjects	Optional. If an unsupported object is included in a request, this parameter indicates what action to take. If it's set to `false`, an error is returned. If it's set to `true`, the object is ignored and no error is returned. See the Unsupported Objects section for reference. The default is `false`.
limit	Optional. Specifies the maximum number of suggested records to return. If a limit isn't specified, 5 records are returned by default. If there are more suggested records than the limit specified, the response body's `hasMoreResults` property is `true`.
networkId	Optional. Specifies one or more unique identifiers for the Experience Cloud sites to return the question to. Specify multiple sites using a comma-separated list. This parameter is only applicable when the parameter `type` equals `question` or parameter `sobject` equals `user`.
q	Required. The user's search query string, properly URL-encoded. Suggestions are returned only if the user's query string meets the minimum length requirements: one character for queries in Chinese, Japanese, Korean, and Thai; three characters for all other languages. Query strings that exceed the maximum length of 255 characters (or 200 consecutive characters without a space break) return an error.
sobject	Required. The objects that the search is scoped to, such as Account or offer__c. If the `sobject` value is `feedItem`, the `type` parameter is required and it must have a value of `question`. Specify up to 10 objects with a comma-separated list. For example: `sobject=Account,Contact,Lead`. To take advantage of the feature, activate the **CrossObjectTypeahead** permission. To specify the specific fields to return by object, use the following syntax with multiple fields in a comma-separated list. The *`sobject`* is lowercase. `sobject=`***`sobject`***`.fields=`***`fields`***

Parameter	Description
	For example: `&sobject=Account,Contact,Lead&account.fields=Website,Phone &contact.fields=Phone`
`topicId`	Optional. Specifies the unique identifier of the single topic that the question to return was tagged as. This parameter is only applicable when the parameter `type` equals `question`.
`type`	Required when the `sobject` value is `feedItem`. Including this parameter for all other `sobject` values doesn't affect the query. Specifies that the type of Feed is questions. Valid value: `question`.
`userId`	Optional. Specifies one or more unique identifiers of one or more users who authored the question to return. Specify multiple users using a comma-separated list. This parameter is only applicable when the parameter type equals `question`. Don't use with the `groupId`.

`useSearchScope`	Optional. Available in API version 40.0 and later. The default value is `false`. If `false`, the objects specified in the request are used to suggest records. If `true`, in addition to the objects specified in the request, the user's search scope is used to suggest records. The search scope is the list of objects a user uses most frequently.
	• If the request doesn't specify an object, use `useSearchScope=true`.
	• If `useSearchScope=true` and the user's search scope is empty, the default search scope is used to suggest records.
	• Typically, only the first 10 objects are used to suggest records. However, an admin can assign objects that are always considered when returning results. If configured, up to 15 objects are used to suggest records. For more information about assigning objects, see Assign Search Results Objects to Users (Beta).
	• Objects specified in the `sobject` parameter are prioritized over objects in the user's search scope.
	• Values for the `ignoreUnsupportedSObjects` parameter aren't applied to the objects in the search scope.
	This example uses only the search scope.
	`.../search/suggestions?q=Acme&useSearchScope=true`
	This example uses the search scope and the Account object.
	`.../search/suggestions?q=Acme&sobject=Account&useSearchScope=true`

`where`	Optional. A filter that follows the same syntax as the SOQL `WHERE` clause. URLs encode the expression. Use the clause for an object, or globally for all compatible objects. An example of an object-specific clause is: `account.where=name%20LIKE%20%27Smith%25%27`. An example of a global clause is: `where=name%20LIKE%20%27Smith%25%27`. The parameter

Parameter	Description
	must be lower case. Any object-specific `where` clauses override the global `where` clause. You can't use this parameter for the Question object. To specify multiple entities, see the following example. This feature is available in version 38.0 and later.

```
...search/suggestions?q=Smith
&sobject=Account,Contact,KnowledgeArticleVersion,CollaborationGroup,Topic,FeedItem

// Specifies a global where clause (to filter Account and
 Contact)
&where=name%20LIKE%20%27Smith%25%27
// Overrides the global where clause for Knowledge Article

(filtering by PublishStatus and Language is required for
 KnowledgeArticle)
&knowledgearticleversion.where=PublishStatus='online'+and+language='en_US'
// Overrides the global where clause for Topic
&topic.where=networkid=<1234567891>
// Overrides the global where clause for
CollaborationGroup
&collaborationgroup.where=networkid=<1234567891>
// FeedItem-Question doesn't support where clauses, but
we can filter
the type and networkId&type=question
&networkId==<1234567891>
```

Example

Example Response Body

```
{
  "autoSuggestResults" : [ {
      "attributes" : {
        "type" : "Account",
        "url" : "/services/data/v60.0/sobjects/Account/001xx000003DH6WAAW"
      },
      "Id" : "001xx000003DH6WAAW",
      "Name" : "National Utility Service"
    }, {
    {
      "attributes" : {
        "type" : "Account",
        "url" : "/services/data/v60.0/sobjects/Account/001xx000003DHJ4AAO"
      },
      "Id" : "001xx000003DHJ4AAO",
      "Name" : "National Utility Service"
    }, {
    {
      "attributes" : {
        "type" : "Account",
```

```
        "url" : "/services/data/v60.0/sobjects/Account/001xx000003DHscAAG"
      },
      "Id" : "001xx000003DHscAAG",
      "Name" : "National Urban Technology Center"
    } ],
  "hasMoreResults" : false,
  "meta" : {
    "nameFields" : [ {
      "entityApiName" : "Account",
      "fieldApiName" : "Name"
    } ],
    "secondaryFields" : [ ]
  }
}
```


Example Response Body for a Multiple Object Request


```
{
  "autoSuggestResults" : [ {
      "attributes" : {
        "type" : "Account",
        "url" : "/services/data/v60.0/sobjects/Account/001xx000003DMEKAA4"
      },
      "Id" : "001xx000003DMEKAA4"
      "Name" : "Joe Doe Printing"
    }, {
    {
      "attributes" : {
        "type" : "Account",
        "url" : "/services/data/v60.0/sobjects/Account/001xx000003DLjvAAG"
      },
      "Id" : "001xx000003DLjvAAGO"
      "Name" : "Joe Doe Plumbing"
    }, {
    {
      "attributes" : {
        "type" : "Contact",
        "url" : "/services/data/v60.0/sobjects/Contact/003xx000004U9Y9AAK"
      },
      "Id" : "003xx000004U9Y9AAK"
      "Name" : "John Doe"
    } ],
  "hasMoreResults" : false,
  "meta" : {
    "nameFields" : [ {
      "entityApiName" : "Account",
      "fieldApiName" : "Name"
    }, {
      "entityApiName" : "Contact",
      "fieldApiName" : "Name"
    } ],
    "secondaryFields" : [ ]
  }
}
```

Example XML Response Body


```
<?xml version="1.0" encoding="UTF-8"?
<suggestions>
  <autoSuggestResults type="Account"
url="/services/data/v60.0/sobjects/Account/001xx000003DH6WAAW">
    <Id>001xx000003DH6WAAW</Id>
    <Name>National Utility Service</Name>
  </autoSuggestResults>
  <autoSuggestResults type="Account"
url="/services/data/v60.0/sobjects/Account/001xx000003DHJ4AAO">
    <Id>001xx000003DHJ4AAO</Id>
    <Name>National Utility Service</Name>
  </autoSuggestResults>
  <autoSuggestResults type="Account"
url="/services/data/v60.0/sobjects/Account/001xx000003DHscAAG">
    <Id>001xx000003DHscAAG</Id>
    <Name>National Urban Technology Center</Name>
  </autoSuggestResults>
  <hasMoreResults>true</hasMoreResults>
  <meta>
    <nameFields>
      <entityApiName>Account</entityApiName>
      <fieldApiName>Name</fieldApiName>
    </nameFields>
    <nameFields>
      <entityApiName>ContentDocument</entityApiName>
      <fieldApiName>Title</fieldApiName>
    </nameFields>
  </meta>
</suggestions>
```


Search Suggested Article Title Matches

Returns a list of Salesforce Knowledge article titles that match the user's search query string. Provides a shortcut to navigate directly to likely relevant articles before the user performs a search. This resource is available in REST API version 30.0 and later.

Salesforce Knowledge must be enabled in your organization. The user must have the "View Articles" permission enabled. The articles suggested include only the articles the user can access, based on the data categories and article types the user has permissions to view.

The Suggest Article Title Matches resource is designed to return display-ready data about likely relevant articles. Articles are suggested if their titles contain the entire query string, except stopwords, such as "a," "for," and "the."

For example, a search for `Backpacking for desert` returns the article, "Backpacking in the desert."

Note: Articles with titles that include stopwords from the query string, such as "Backpacking for desert survival" in this example, appear before matching articles with titles that don't include the stopwords.

Stopwords at the end of the query string are treated as search terms.

A wildcard is automatically appended to the last token in the query string.

Note: If the user's search query contains quotation marks or wildcards, those symbols are automatically removed from the query string in the URI along with any other special characters.

If the number of suggestions returned exceeds the limit specified in the request, the end of the response contains a field called `hasMoreResults`. Its value is `true` if the suggestions returned are only a subset of the suggestions available, and `false` otherwise.

Syntax

URI

`/services/data/vXX.X/search/suggestTitleMatches?q=searchString&language=articleLanguage&publishStatus=articlePublicationStatus`

Formats

JSON, XML

HTTP methods

GET

Authentication

`Authorization: Bearer token`

Request body

None required

Request parameters

Parameter	Description
`articleTypes`	Optional. Three-character ID prefixes indicating the desired article types. You can specify multiple values for this parameter in a single REST call, by repeating the parameter name for each value. For example, `articleTypes=ka0&articleTypes=ka1`.
`categories`	Optional. The name of the data category group and name of the data category for desired articles, expressed as a JSON mapping. You can specify multiple data category group and data category pairs in this parameter. For example, `categories={"Regions":"Asia","Products":"Laptops"}`. Characters in the URL might need to be encoded. For this example, `categories=%7B%22Regions%22%3A%22Asia%22%2C%22Products%22%3A%22Laptops%22%7D`.
`channel`	Optional. The channel where the matching articles are visible. Valid values: • `AllChannels`–Visible in all channels the user has access to • `App`–Visible in the internal Salesforce Knowledge application • `Pkb`–Visible in the public knowledge base • `Csp`–Visible in the Customer Portal • `Prm`–Visible in the Partner Portal If `channel` isn't specified, the default value is determined by the type of user. • `Pkb` for a guest user • `Csp` for a Customer Portal user • `Prm` for a Partner Portal user • `App` for any other type of user If `channel` is specified, the specified value may not be the actual value requested, because of certain requirements.

Parameter	Description
	• For guest, Customer Portal, and Partner Portal users, the specified value must match the default value for each user type. If the values don't match or `AllChannels` is specified, then `App` replaces the specified value. • For all users other than guest, Customer Portal, and Partner Portal users: – If `Pkb`, `Csp`, `Prm`, or `App` are specified, then the specified value is used. – If `AllChannels` is specified, then `App` replaces the specified value.
`language`	Required. The language of the user's query. Specifies the language that matching articles are written in.
`limit`	Optional. Specifies the maximum number of articles to return. If there are more suggested articles than the limit specified, the response body's `hasMoreResults` property is `true`.
`publishStatus`	Required. The article's publication status. Valid values: • `Draft`—Articles aren't published in Salesforce Knowledge. • `Online`—Articles are published in Salesforce Knowledge. • `Archived`—Articles aren't published and are available in Archived Articles view.
`q`	Required. The user's search query string, properly URL-encoded. Suggestions are returned only if the user's query string meets the minimum length requirements: one character for queries in Chinese, Japanese, and Korean, and three characters for all other languages. Query strings exceeding the maximum length of 250 characters return an error.
`topics`	Optional. The topic of the returned articles. For example: `topics=outlook&topics=email`.
`validationStatus`	Optional. The validation status of returned articles.

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/search/suggestTitleMatches?
q=orange+banana&language=en_US&publishStatus=Online -H "Authorization: Bearer token"
```

Example Response Body

```
{
  "autoSuggestResults" : [ {
    "attributes" : {
    "type" : "KnowledgeArticleVersion",
    "url" : "/services/data/v60.0/sobjects/KnowledgeArticleVersion/ka0D00000004CcQ"
    },
  "Id" : "ka0D00000004CcQ",
  "UrlName" : "orange-banana",
```

```
  "Title" : "orange banana",
  "KnowledgeArticleId" : "kA0D00000004Cfz"
  } ],
  "hasMoreResults" : false
}
```


Search Suggested Queries

Returns a list of suggested searches based on the user's query string text matching searches that other users have performed in Salesforce Knowledge. Provides a way to improve search effectiveness, before the user performs a search. This resource is available in REST API version 30.0 and later.

Salesforce Knowledge must be enabled in your organization.

Queries are suggested if they exactly match the query string text. The text string must be a prefix within the query; it's not considered a match if it appears within a word. For example, the text string `app` would return suggested queries *apple banana* and *banana apples* but not *pineapple*.

If the number of suggestions returned exceeds the limit specified in the request, the end of the response contains a field called `hasMoreResults`. Its value is `true` if the suggestions returned are only a subset of the suggestions available, and `false` otherwise.

If the user's search query contains quotation marks or wildcards, those symbols are automatically removed from the query string in the URI.

Syntax

URI

`/services/data/vXX.X/search/suggestSearchQueries?q=searchString&language=languageOfQuery`

Formats

JSON, XML

HTTP methods

GET

Authentication

`Authorization: Bearer token`

Request body

None required

Request parameters

Parameter	Description
`channel`	Optional. Specifies the Salesforce Knowledge channel where the article can be viewed. Valid values: • `AllChannels`–Visible in all channels the user has access to • `App`–Visible in the internal Salesforce Knowledge application • `Pkb`–Visible in the public knowledge base • `Csp`–Visible in the Customer Portal • `Prm`–Visible in the Partner Portal

Parameter	Description
	If `channel` isn't specified, the default value is determined by the type of user. • `Pkb` for a guest user • `Csp` for a Customer Portal user • `Prm` for a Partner Portal user • `App` for any other type of user If `channel` is specified, the specified value may not be the actual value requested, because of certain requirements. • For guest, Customer Portal, and Partner Portal users, the specified value must match the default value for each user type. If the values don't match or `AllChannels` is specified, then `App` replaces the specified value. • For all users other than guest, Customer Portal, and Partner Portal users: – If `Pkb`, `Csp`, `Prm`, or `App` are specified, then the specified value is used. – If `AllChannels` is specified, then `App` replaces the specified value.
`language`	Required. The language of the user's query.
`limit`	Optional. Specifies the maximum number of suggested searches to return. If there are more suggested queries than the limit specified, the response body's `hasMoreResults` property is `true`.
`q`	Required. The user's search query string, properly URL-encoded. Suggestions are returned only if the user's query string meets the minimum length requirements: one character for queries in Chinese, Japanese, and Korean, and three characters for all other languages. Query strings exceeding the maximum length of 250 characters return an error.

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/search/suggestSearchQueries?
q=app&language=en_US -H "Authorization: Bearer token"
```

Example Response Body

```
{
  "autoSuggestResults" : [ {
    "0" : "apple",
    "1" : "apple banana",
  } ],
  "hasMoreResults" : false
}
```

Salesforce Surveys Translation Resources

Use REST APIs to translate survey fields, view, update, or delete translated survey fields. The translated values of surveys fields are stored in Flow fields.

The following survey fields can be translated:

- Survey name
- Pause message
- Welcome message
- Questions
- Answer choices and ranking items
- Thank you message

IN THIS SECTION:

Add or Change the Translation of a Survey Field

If a survey field can be translated or is already translated into a particular language, you can add or change the translated value of the survey field. This resource is available in REST API version 48.0 and later.

Note: This URI can only be used for the flow process type `Survey`.

Syntax

URI

`/services/data/vXX.X/localizedvalue/record/developerName/language`

Formats

JSON

HTTP methods

POST

Authentication

`Authorization: Bearer token`

Request body JSON example

```
{
"value": "China"
}
```

Request parameters

Parameter	Description
developerName	Optional. Developer name of the flow field.
language	Optional Translated language of the flow field.
value	Required. Translated value of the flow field.

Response parameters

Parameter	Description
createdBy	ID of the user who translated the flow field.
createdDate	Date and time the flow field was translated.
developerName	Developer name of the flow field.
language	Language into which the flow field was translated.
value	Translated value of the flow field.
isOutofDate	Indicates if the flow field is out of date.

Example

```
{
  "createdBy": "005xxx",
  "createdDate": "2018-09-14T00:10:30Z",
  "developerName": "Flow.Flow.MyFlow.1.Choice.Choice_1_Master.InputLabel",
  "language": "zh_CN",
  "value": "□□",
  "isOutOfDate": true
}
```

Add or Update the Translated Value of Multiple Survey Fields in One or More Languages

If one or more survey fields can be translated or are already translated, you can add or update the translated values of the survey fields in the languages into which survey fields can be translated. This resource is available in REST API version 48.0 and later.

Note: This URI can only be used for the flow process type `Survey`.

Syntax

URI

`/services/data/vXX.X/localizedvalue/records/upsert`

Formats

JSON

HTTP methods

POST

Request body JSON example

```
[
  {
    "developerName": "Flow.Flow.MyFlow.1.Choice.Choice_1_Master.InputLabel",
    "language": "en_US",
    "value": "China"
  },
  {
    "developerName": "Flow.Flow.MyFlow.1.Choice.Choice_1_Master.InputLabel",
    "language": "zh_CN",
    "value": "□□"
  }
]
```

Request parameters

Parameter	Description
developerName	Required. Developer name of the flow field.
language	Required. Language into which the flow field is translated.
value	Required. New or updated value of the flow field.

Response parameters

Parameter	Description
createdBy	ID of the user who translated the flow field.
createdDate	Date and time the flow field was translated.
developerName	Developer name of the flow field.

Parameter	Description
`language`	Language into which the flow field was translated.
`value`	Updated value of the flow field.
`isOutofDate`	Indicates if the flow field is out of date.

Example

```
[
  {
    "createdBy": "005xxx",
    "createdDate": "2018-09-14T00:10:30Z",
    "developerName": "Flow.Flow.MyFlow.1.Choice.Choice_1_Master.InputLabel",
    "language": "en_US",
    "value": "China",
    "isOutOfDate": false
  },
  {
    "createdBy": "005xxx",
    "createdDate": "2018-09-14T00:10:30Z",
    "developerName": "Flow.Flow.MyFlow.1.Choice.Choice_1_Master.InputLabel",
    "language": "zh_CN",
    "value": "□□",
    "isOutOfDate": false
  }
]
```

Delete the Translated Value of a Survey Field

After a survey field is translated into a particular language, you can delete the translated value of the survey field. This resource is available in REST API version 48.0 and later.

Note: This URI can only be used for the flow process type `Survey`.

Syntax

URI

`/services/data/vXX.X/localizedvalue/record/developerName/language`

Formats

JSON

HTTP methods

DELETE

Request parameters

Parameter	Description
`developerName`	The developer name of the flow field. For example, *`Flow.Flow.MyFlow.1.Choice.Choice_1_Master.InputLabel`*

Parameter	Description
`language`	Language of the translated field. Possible values are: • `da` • `nl_NL` • `fi` • `fr` • `de`

Delete the Translated Value of Multiple Survey Fields in One or More Languages

After survey fields are translated into one or more languages, you can delete the translated values of multiple survey fields. This resource is available in REST API version 48.0 and later.

Note: This URI can only be used for the flow process type `Survey`.

Syntax

URI

`/services/data/vXX.X/localizedvalue/records/delete`

Formats

JSON

HTTP methods

POST

Request body JSON example

```
[
  {
    "developerName": "Flow.Flow.MyFlow.1.Choice.Choice_1_Master.InputLabel",
    "language": "en_US"
  },
  {
    "developerName": "Flow.Flow.MyFlow.1.Choice.Choice_1_Master.InputLabel",
    "language": "zh_CN"
  }
]
```

Request parameters

Parameter	Description
`developerName`	Required. Developer name of the flow field.
`language`	Required. Language into which the flow field was translated.

Get Translated Value of a Survey Field

After a survey field is translated into a particular language, you can use this resource to get the translated value of the survey field. This resource is available in REST API version 48.0 and later.

Note: This URI can only be used for the flow process type `Survey`.

Syntax

URI

`/services/data/vXX.X/localizedvalue/record/developerName/language`

Formats

JSON

HTTP methods

GET

Authentication

`Authorization: Bearer token`

Request body

None

Request parameters

Path Parameter	Description
`developerName`	Required. The developer name of the flow field. For example, *`Flow.Flow.MyFlow.1.Choice.Choice_1_Master.InputLabel`*
`language`	Required. Language of the translated field. Possible values are: • `da` • `nl_NL` • `fi` • `fr` • `de`

Response parameters

Parameter	Description
`createdBy`	ID of the user who translated the flow field.
`createdDate`	Date and time the flow field was translated.
`developerName`	Developer name of the flow field.
`language`	Language into which the flow field was translated.
`value`	Translated value of the flow field.
`isOutofDate`	Indicates if the flow field is out of date.

Example

```
{
  "createdBy": "005xxx",
  "createdDate": "2018-09-14T00:10:30Z",
  "developerName": "Flow.Flow.MyFlow.1.Choice.Choice_1_Master.InputLabel",
  "language": "zh_CN",
  "value": "□□",
  "isOutOfDate": true
}
```

Get the Translated Values of Multiple Survey Fields in One or More Languages

After survey fields are translated into one or more languages, you can view the translated values of multiple survey fields in the translated languages. This resource is available in REST API versions 48.0 and later.

Note: This URI can only be used for the flow process type `Survey`.

Syntax

URI

`/services/data/vXX.X/localizedvalue/records/get`

Formats

JSON

HTTP methods

POST

Request body JSON example

```
[
  {
     "developerName": "Flow.Flow.MyFlow.1.Choice.Choice_1_Master.InputLabel",
     "language": "en_US"
  },
  {
     "developerName": "Flow.Flow.MyFlow.1.Choice.Choice_1_Master.InputLabel",
     "language": "zh_CN"
  }
]
```

Request parameters

Parameter	Description
`developerName`	Required. Developer name of the flow field.
`language`	Required. Language into which the flow field was translated.

Response parameters

Parameter	Description
`createdBy`	ID of the user who translated the flow field.
`createdDate`	Date and time the flow field was translated.
`developerName`	Developer name of the flow field.
`language`	Language into which the flow field was translated.
`value`	Translated value of the flow field.
`isOutofDate`	Indicates if the flow field is out of date.

Example

```
[
  {
    "createdBy": "005xxx",
    "createdDate": "2018-09-14T00:10:30Z",
    "developerName": "Flow.Flow.MyFlow.1.Choice.Choice_1_Master.InputLabel",
    "language": "en_US",
    "value": "China",
    "isOutOfDate": true
  },
  {
    "createdBy": "005xxx",
    "createdDate": "2018-09-14T00:10:30Z",
    "developerName": "Flow.Flow.MyFlow.1.Choice.Choice_1_Master.InputLabel",
    "language": "zh_CN",
    "value": "□□",
    "isOutOfDate": true
  }
]
```

Tabs

Returns a list of all tabs—including Lightning page tabs—available to the current user, regardless of whether the user has chosen to hide tabs via the All Tabs (+) tab customization feature. This resource is available in REST API version 31.0 and later.

IN THIS SECTION:

Get Tabs

Gets a list of all tabs—including Lightning page tabs—available to the current user, regardless of whether the user has chosen to hide tabs via the All Tabs (**+**) tab customization feature. This resource is available in REST API version 31.0 and later.

Syntax

URI
: `/services/data/vXX.X/tabs/`

Formats
: JSON, XML

HTTP methods
: GET

Authentication
: `Authorization: Bearer token`

Request body
: None

Request parameters
: None

Example

Example Request

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/tabs -H "Authentication:
 Bearer token"
```

Example Response Body

This is a partial code sample, representing the Accounts tab.

```
[...,
  "colors" : [ {
    "color" : "6f7ccb",
    "context" : "primary",
    "theme" : "theme4"
  }, {
    "color" : "236FBD",
    "context" : "primary",
    "theme" : "theme3"
  } ],
  "custom" : false,
  "iconUrl" : "https://MyDomainName.my.salesforce.com/img/icon/accounts32.png",
  "icons" : [ {
    "contentType" : "image/png",
    "height" : 32,
    "theme" : "theme3",
    "url" : "https://MyDomainName.my.salesforce.com/img/icon/accounts32.png",
    "width" : 32
  }, {
    "contentType" : "image/png",
```

```
    "height" : 16,
    "theme" : "theme3",
    "url" : "https://MyDomainName.my.salesforce.com/img/icon/accounts16.png",
    "width" : 16
  }, {
    "contentType" : "image/svg+xml",
    "height" : 0,
    "theme" : "theme4",
   "url" : "https://MyDomainName.my.salesforce.com/img/icon/t4/standard/account.svg",

    "width" : 0
  }, {
    "contentType" : "image/png",
    "height" : 60,
    "theme" : "theme4",
   "url" : "https://MyDomainName.my.salesforce.com/img/icon/t4/standard/account_60.png",

    "width" : 60
  }, {
    "contentType" : "image/png",
    "height" : 120,
    "theme" : "theme4",
    "url" :
"https://MyDomainName.my.salesforce.com/img/icon/t4/standard/account_120.png",
    "width" : 120
  } ],
  "label" : "Accounts",
  "miniIconUrl" : "https://MyDomainName.my.salesforce.com/img/icon/accounts16.png",
  "name" : "standard-Account",
  "sobjectName" : "Account",
  "url" : "https://MyDomainName.my.salesforce.com/001/o",
...]
```


Return Headers Using Tabs

Returns only the headers that are returned by a GET request to Tabs resources. This gives you a chance to see header values before retrieving the content of the resource. This resource is available in REST API version 31.0 and later.

Syntax

URI

`/services/data/vXX.X/tabs/`

Formats

JSON, XML

HTTP methods

HEAD

Authentication

`Authorization: Bearer token`

Request body

None

Request parameters
None

Example

Return headers of request for all tabs

```
curl -X HEAD --head https://MyDomainName.my.salesforce.com/services/data/v60.0/tabs -H
 "Authorization: Bearer token"
```

Themes

Gets the list of icons and colors used by themes in the Salesforce application. Theme information is provided for objects in your organization that use icons and colors in the Salesforce UI. This resource is available in REST API version 29.0 and later.

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

Syntax

URI
`/services/data/vXX.X/theme`

Formats
JSON, XML

HTTP methods
GET

Authentication
`Authorization: Bearer token`

Request body
None

Request parameters
None

Response data
An array of theme items. Each theme item contains the following fields:

Name	Type	Description
`colors`	array of theme colors	Array of theme colors.
`icons`	array of theme icons	Array of theme icons.
`name`	string	Name of the object that the theme colors and icons are associated with.

Each theme color contains the following fields:

Name	Type	Description
`color`	string	The color described in Web color RGB format, for example, "00FF00".
`context`	string	The color context, which determines whether the color is the main color ("primary") for the object, or not.
`theme`	string	The associated theme. Possible values include: • *theme2*—Theme used prior to Spring '10, called the "Salesforce Classic 2005 user interface theme" • *theme3*—Theme introduced in Spring '10, called the "Salesforce Classic 2010 user interface theme" • *theme4*—Theme introduced in Winter '14 for the mobile touchscreen version of Salesforce, and in Winter '16 for Lightning Experience • *custom*—Theme associated with a custom icon

Each theme icon contains the following fields:

Name	Type	Description
`contentType`	string	The icon's content type, for example, "image/png."
`height`	number	The icon's height in pixels. If the icon content type is an SVG type, height and width values are not used.
`theme`	string	The associated theme. Possible values include: • *theme2*—Theme used prior to Spring '10, called the "Salesforce Classic 2005 user interface theme" • *theme3*—Theme introduced in Spring '10, called the "Salesforce Classic 2010 user interface theme" • *theme4*—Theme introduced in Winter '14 for the mobile touchscreen version of Salesforce, and in Winter '16 for Lightning Experience • *custom*—Theme associated with a custom icon
`url`	string	The fully qualified URL for this icon.
`width`	number	The icon's width in pixels. If the icon content type is an SVG type, height and width values are not used.

Example

```
{
    "themeItems" : [
    {
        "name" : "Merchandise__c",
        "icons" : [
        {
```

```
            "contentType" : "image/png",
            "width" : 32,
            "url" : "https://MyDomainName.my.salesforce.com/img/icon/computer32.png",
            "height" : 32,
            "theme" : "theme3"
        },
        {
            "contentType" : "image/png",
            "width" : 16,
            "url" : "https://MyDomainName.my.salesforce.com/img/icon/computer16.png",
            "height" : 16,
            "theme" : "theme3"
        } ],
        "colors" : [
        {
            "context" : "primary",
            "color" : "6666CC",
            "theme" : "theme3"
        },
        {
            "context" : "primary",
            "color" : "66895F",
            "theme" : "theme4"
        },
    ...
    }
...
}
```

Composite

Executes a series of REST API requests in a single POST request, or retrieves a list of other composite resources with a GET request.

IN THIS SECTION:

Send Multiple Requests Using Composite

Executes a series of REST API requests in a single call. You can use the output of one request as the input to a subsequent request. The response bodies and HTTP statuses of the requests are returned in a single response body. The entire series of requests counts as a single call toward your API limits.

Get a List of Composite Resources

Gets a list of URIs for other composite resources.

Send Multiple Requests Using Composite

Executes a series of REST API requests in a single call. You can use the output of one request as the input to a subsequent request. The response bodies and HTTP statuses of the requests are returned in a single response body. The entire series of requests counts as a single call toward your API limits.

The requests in a composite call are called *subrequests*. All subrequests are executed in the context of the same user. In a subrequest's body, you specify a reference ID that maps to the subrequest's response. You can then refer to the ID in the `url` or `body` fields of later subrequests by using a JavaScript-like reference notation.

For example, the following composite request body includes two subrequests. The first creates an account and designates the output as `refAccount`. The second creates a contact parented under the new account by referencing `refAccount` in the subrequest body.

```
{
"compositeRequest" : [{
  "method" : "POST",
  "url" : "/services/data/v60.0/sobjects/Account",
  "referenceId" : "refAccount",
  "body" : { "Name" : "Sample Account" }
  },{
  "method" : "POST",
  "url" : "/services/data/v60.0/sobjects/Contact",
  "referenceId" : "refContact",
  "body" : {
    "LastName" : "Sample Contact",
    "AccountId" : "@{refAccount.id}"
    }
  }]
}
```

You can specify whether an error in a subrequest causes the whole composite request to roll back or just the subrequests that depend on it. You can also specify headers for each subrequest.

Composite is supported for the following resources.

- All sObject resources (`/services/data/vXX.X/sobjects/`), including sObject Rows by External ID on page 152 and excluding sObject Blob Get
- The Query resource (`/services/data/vXX.X/query/?q=soql`)
- The QueryAll resource (`/services/data/vXX.X/queryAll/?q=soql`)
- The sObject Collections resource (`/services/data/vXX.X/composite/sobjects`). Available in API version 43.0 and later.

Note: You can have up to 25 subrequests in a single call. Up to 5 of these subrequests can be sObject Collections or query operations, including Query and QueryAll requests.

Syntax

URI

`/services/data/vXX.X/composite`

Formats

JSON

HTTP method

POST

Authentication

`Authorization: Bearer token`

Parameters

None required

Request body

Composite Request Body

Response body

Composite Response Body

Example

For examples of using the Composite resource, see Execute Dependent Requests in a Single API Call and Update an Account, Create a Contact, and Link Them with a Junction Object.

IN THIS SECTION:

Composite Subrequest Result

The composite subrequest result describes the result for a subrequest.

Composite Request Body

Describes a collection of subrequests to execute with the Composite resource.

Composite Collection Input

The request body contains an `allOrNone` flag that specifies how to roll back errors and a `compositeRequest` collection that includes subrequests to execute.

Properties

Name	Type	Description	Required or Optional
`allOrNone`	Boolean	Specifies what to do when an error occurs while processing a subrequest. If the value is `true`, the entire composite request is rolled back. The top-level request returns HTTP 200 and includes responses for each subrequest. If the value is `false`, the remaining subrequests that don't depend on the failed subrequest are executed. Dependent subrequests aren't executed. In either case, the top-level request returns HTTP 200 and includes responses for each subrequest. Note: If the Composite request includes an sObject Collections request, the sObject Collections request's `allOrNone` parameter can also affect the results. See allOrNone Parameters in Composite and Collections Requests.	Optional
`collateSubrequests`	Boolean	Controls whether the API collates unrelated subrequests to bulkify them (`true`) or not (`false`).	Optional

Name	Type	Description	Required or Optional
		When subrequests are collated, the processing speed is faster, but the order of execution is not guaranteed (unless there is an explicit dependency between the subrequests). If collation is disabled, then the subrequests are executed in the order in which they are received. Subrequests that contain valid HTTP headers are not collated. In API version 49.0 and later, the default value is `true`. In version 48.0, the default value is `false`. Subrequests can be collated if they meet these conditions: • The HTTP methods are the same. • The API versions of the resources are the same. • The parents of the resources are `/sobjects` resources. • No more than five sObjects resources are present across a grouping of subrequests. Note: Collation can cause issues if there are implicit but not explicit dependencies between items. For example, consider a request that creates • an Account • a Contact related to the Account • a custom object that has a trigger dependent on the account name. The Account and the custom object are collated, since there is no explicit dependency. But the custom object's trigger may fail because there is no guarantee that the Account is created first. If you have relationships like this where you need to control the order of execution, set `collateSubrequests` to `false`. Available in API version 48.0 and later. (In earlier versions, subrequests cannot be collated.)	
`compositeRequest`	Composite Subrequest[]	Collection of subrequests to execute.	Required

JSON example

```
{
   "allOrNone" : true,
   "collateSubrequests": true,
   "compositeRequest" : [{
      Composite Subrequest
```

```
        },{
        Composite Subrequest
        },{
        Composite Subrequest
    }]
}
```

Composite Subrequest

Contains the resource, method, headers, body, and reference ID for the subrequest.

Properties

Name	Type	Description	Required or Optional
`body`	Varies	The input body for the subrequest. The type depends on the request specified in the `url` property. The referenceable types are DateTime, String, Boolean, Byte, Character, Short, Integer, Long, Double, and Float.	Optional
`httpHeaders`	Map<String, String>	Request headers and their values to include with the subrequest. You can include any header supported by the requested resource except for the following three headers. • Accept • Authorization • Content-Type Subrequests inherit these three header values from the top-level request. Don't specify these headers in a subrequest. If you do, the top-level request fails and returns an HTTP 400 response. Note: Subrequests that contain valid HTTP headers cannot be collated, which slows the processing speed of the request.	Optional
`method`	String	The method to use with the requested resource. Possible values are `POST`, `PUT`, `PATCH`, `GET`, and `DELETE` (case-sensitive). For a list of valid methods, refer to the documentation for the requested resource.	Required
`referenceId`	String	Reference ID that maps to the subrequest's response and can be used to reference the response in later subrequests. You can reference the `referenceId` in either the body or URL of a subrequest. Use this syntax to include a reference: `@{referenceId.FieldName}`. You can use two operators with the reference ID. The `.` operator references a field on a JSON object in the response. For example, let's say you retrieve an account record's data in one subrequest and assign the reference ID	Required

Name	Type	Description	Required or Optional
		`Account1Data` to the output. You can refer to the account's name in later subrequests like this: `@{Account1Data.Name}`. The `[]` operator indexes a JSON collection in the response. For example, let's say you request basic account information with the sObject Basic Information resource in one subrequest and assign the reference ID `AccountInfo` to the output. Part of the response includes a collection of recently created accounts. You can refer to the ID of the first account in the recent items collection like this: `@{AccountInfo.recentItems[0].Id}`. You can use each operator recursively as long as it makes sense in the context of the response. For example, to refer to the billing city component of an account's compound address field: `@{NewAccount.BillingAddress.city}`. `referenceId` is case-sensitive, so pay close attention to the case of the fields you're referring to. See Usage. Note: • The `referenceId` must start with a letter or a number. • The `referenceId` must not contain anything besides letters, numbers, or underscores ('_').	
`url`	String	The resource to request. • The URL can include any query string parameters that the subrequest supports. The query string must be URL-encoded. • You can use parameters to filter response bodies. • The URL must start with `/services/data/vXX.X/`.	Required

JSON examples

Example 1

```
{
   "method" : "GET",
   "url" : "/services/data/v60.0/sobjects/Account/describe",
   "httpHeaders" : { "If-Modified-Since" : "Tue, 31 May 2016 18:00:00 GMT" },
   "referenceId" : "AccountInfo"
}
```

Example 2

```
{
   "method" : "POST",
```

```
    "url" : "/services/data/v60.0/sobjects/Account",
    "referenceId" : "refAccount",
    "body" : { "Name" : "Sample Account" }
}
```

Example 3

```
{
    "method" : "GET",
    "url" : "/services/data/v60.0/sobjects/Account/@{refAccount.id}",
    "referenceId" : "NewAccountFields"
}
```

Example 4

```
{
    "method" : "PATCH",
    "url" : "/services/data/v60.0/sobjects/Account/ExternalAcctId__c/ID12345",
    "referenceID" : "NewAccount",
    "body" : { "Name" : "Acme" }
}
```

Usage

Because `referenceId` is case-sensitive, it's important to note the case of the fields that you're referring to. The same field can use different cases in different contexts. For example, when you create a record, the ID field appears as `id` in the response. But when you access a record's data with the sObject Rows resource, the ID field appears as `Id`. In Example 3, the `@{refAccount.id}` reference is valid because `refAccount` refers to the response from a POST like that shown in Example 2. If you use `Id` instead (mixed case rather than all lowercase), as in `@{refAccount.Id}`, you get an error when sending the request because the reference ID uses the wrong case.

Note: You can have up to 25 subrequests in a single call. Up to 5 of these subrequests can be sObject Collections or query operations, including Query and QueryAll requests.

Composite Response Body

Describes the result of a Composite request.

Composite Results

Properties

Name	Type	Description
compositeResponse	Composite Subrequest Result on page 366[]	Collection of subrequest results

JSON example

```
{
    "compositeResponse" : [{
        Composite Subrequest Result
        },{
        Composite Subrequest Result
```

```
        },{
        Composite Subrequest Result
    }]
}
```

Composite Subrequest Result

The composite subrequest result describes the result for a subrequest.

Properties

Name	Type	Description
`body`	The type depends on the response type of the subrequest.	The response body of this subrequest. See the documentation for the subrequest resource for more information. If the subrequest returns an error, the body includes the error code and message. For more details on error responses, see Status Codes and Error Responses.
`httpHeaders`	Map<String, String>	Response headers and their values for this subrequest. The Composite resource doesn't support the Content-Length header, so subrequest responses don't include this header and neither does the top-level response.
`httpStatusCode`	Integer	An HTTP status code for this subrequest. If `allOrNone` is set to true in the composite request and a subrequest returns an error, all other subrequests return the 400 HTTP status code.
`referenceId`	String	The reference ID specified in the subrequest. This property lets you easily associate subrequests with their results.

Example

```
{
   "body": {
      "id": "001R00000064wdtIAA",
      "success": true,
      "errors": []
   },
   "httpHeaders": {
      "Location": "/services/data/v60.0/sobjects/Account/001R00000064wdtIAA"
   },
```

```
    "httpStatusCode": 201,
    "referenceId": "refAccount"
}
```

The following example shows the response for a subrequest that had an error while trying to create a Contact:

```
{
  "body" : [ {
    "message" : "Email: invalid email address: Not a real email address",
    "errorCode" : "INVALID_EMAIL_ADDRESS",
    "fields" : [ "Email" ]
  } ],
  "httpHeaders" : { },
  "httpStatusCode" : 400,
  "referenceId" : "badContact"
}
```

Behavior and Responses If There Are Illegal Characters in Reference IDs

The `referenceIds` must not contain anything besides letters, numbers, and underscores ('_').

The API's behavior if there are illegal characters depends on the API version and the release. (The pertinent API version is that used to make the composite request. That version isn't necessarily the same as the API version specified in the `url` parameters in the subrequests.)

For example, consider the following request. It attempts to do the following:

- Create an account named "Cloudy Consulting".
- Create a Contact, "Mary Smith", linked to the Cloudy Consulting account.
- Create another new account, named "Easy Spaces".

It has illegal characters in the first reference ID, `refNewAccount[1]`.

```
{
  "allOrNone": false,
  "compositeRequest": [
    {
      "method": "POST",
      "body": {
        "name": "Cloudy Consulting"
      },
      "url": "/services/data/vXX.X/sobjects/Account/",
      "referenceId": "refNewAccount[1]"
    },
    {
      "method": "POST",
      "body": {
        "AccountId": "@{refNewAccount[1].id}",
        "FirstName": "Mary",
        "LastName": "Smith"
      },
      "url": "/services/data/vXX.X/sobjects/Contact",
      "referenceId": "refNewContact"
    },
    {
      "method": "POST",
```

```
      "body": {
        "name": "Easy Spaces"
      },
      "url": "/services/data/vXX.X/sobjects/Account/",
      "referenceId": "refNewAccount2"
    }
  ]
}
```

Version 51.0 and Earlier

In API version 51.0 and earlier, illegal characters in a reference ID cause all the subrequests that use that reference ID to fail. In this example, the response is:

```
HTTP/1.1 200 OK
{
   "compositeResponse" : [
      {
         "body" : {
            "id" : "001R0000006hfeZIAQ",
            "success" : true,
            "errors" : [ ]
         },
         "httpHeaders" : {
            "Location" : "/services/data/v51.0/sobjects/Account/001R0000006hfeZIAQ"
         },
         "httpStatusCode" : 201,
         "referenceId" : "refNewAccount[1]"
      },
      {
         "body" : [
            {
               "errorCode" : "PROCESSING_HALTED",
               "message" : "Invalid reference specified. No value for refNewAccount[1].id
 found in refNewAccount.
            }
         ],
         "httpHeaders" : { },
         "httpStatusCode" : 400,
         "referenceId" : "refNewContact"
      },
      {
         "body" : {
            "id" : "001R0000006hfeeIAA",
            "success" : true,
            "errors" : [ ]
         },
         "httpHeaders" : {
            "Location" : "/services/data/v51.0/sobjects/Account/001R0000006hfeeIAA"
         },
         "httpStatusCode" : 201,
         "referenceId" : "refNewAccount2"
      }
   ]
}
```

The two accounts are created (even though the first account uses illegal characters in its reference ID). But the attempt to create a contact (using the reference ID containing illegal characters) fails.

Responses for Version 51.0 and Earlier in Previous Releases

The response shown above is that for the Summer '21 release and later. In releases before Summer '21, problematic reference IDs in the response were truncated if the IDs contained '(' or '['. So the response would have been:

```
{
   "compositeResponse" : [
      {
         "body" : {
            "id" : "001R0000006hfeZIAQ",
            "success" : true,
            "errors" : [ ]
         },
         "httpHeaders" : {
            "Location" : "/services/data/v51.0/sobjects/Account/001R0000006hfeZIAQ"
         },
         "httpStatusCode" : 201,
         "referenceId" : "refNewAccount"
      },
     ...
}
```

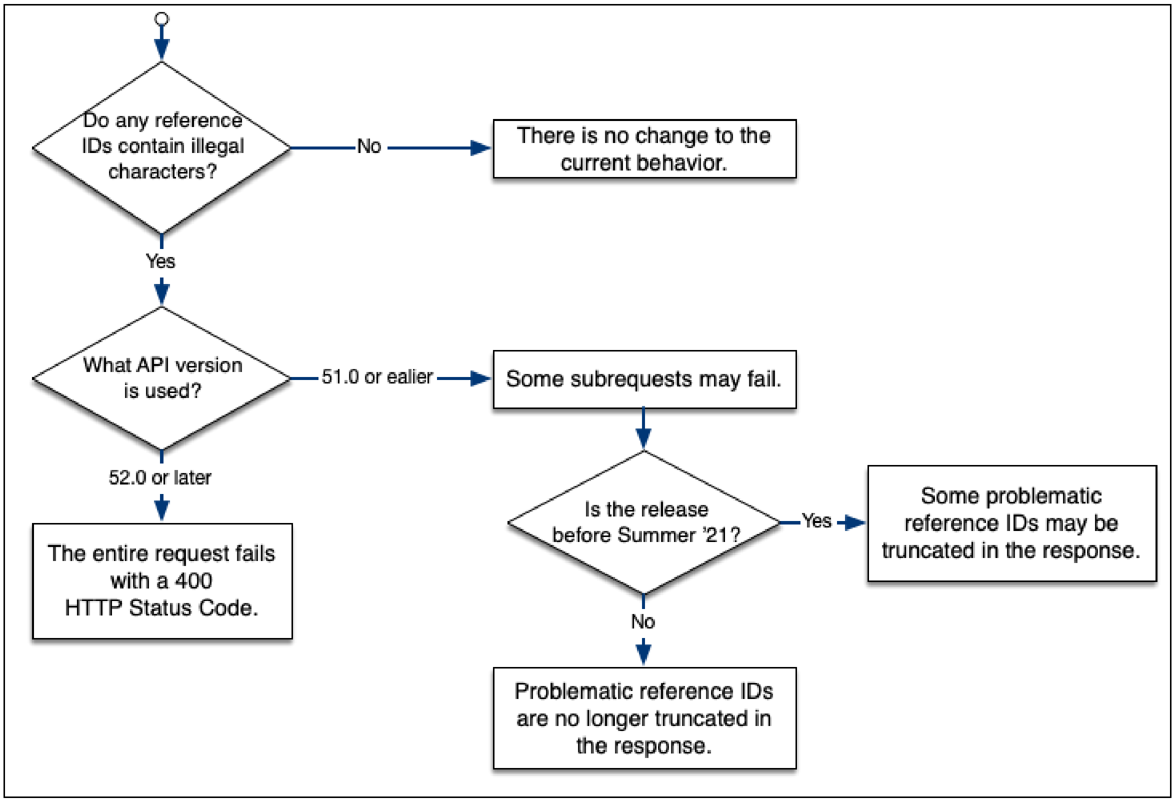
Starting with the Summer '21 release, problematic reference IDs are no longer truncated. This change makes it easier to match the parts of the response to the request.

Version 52.0 and Later

In API version 52.0 and later, any illegal characters in reference IDs cause the whole request to fail. The response to the example above is:

```
HTTP/1.1 400 Bad Request
[
   {
      "message" : "Provided referenceId ('refNewAccount[1]') must start with a letter or
a number, and it can contain only letters, numbers and underscores ('_').",
      "errorCode" : "JSON_PARSER_ERROR"
   }
]
```

Summary



Behavior for References to Null Fields

The API's behavior if there are references to null fields depends on the API version. (The pertinent API version is that used to make the composite request. That version isn't necessarily the same as the API version specified in the `url` parameters in the subrequests.)

Note: This behavior only applies to requests where the parent request uses an sObject Rows resource (for example, `/services/data/vXX.X/sobjects/Contact/id`).

For example, consider this request. It locates an existing Contact and then uses `@{refContact.FirstName}` and `@{refContact.LastName}` to create a record.

```
POST https://MyDomainName.my.salesforce.com/services/data/vXX.X/composite -H "Authorization:
 Bearer token"

   "compositeRequest" : [
      {
         "method" : "GET",
         "url" :
"/services/data/v51.0/sobjects/Contact/003RO0000016kOuYAI?fields=FirstName,LastName",
         "referenceId" : "refContact"
      },
      {
         "method" : "POST",
         "url" : "/services/data/v51.0/sobjects/Contact",
         "referenceId" : "newContact",
         "body" : {
            "FirstName" : "@{refContact.FirstName}",
            "LastName" : "@{refContact.LastName}",
```

```
            "AccountId" : "001RO000001nGCdYAM"
         }
      }
   ]
}
```

What happens if the Contact's first name is null (not set)?

Responses for Version 51.0 and Earlier

In API version 51.0 and earlier, the fact that the Contact's FirstName field is null causes the dependent subrequest to fail.


```
{
   "compositeResponse" : [
      {
         "body" : {
            "attributes" : {
               "type" : "Contact",
               "url" : "/services/data/v51.0/sobjects/Contact/003RO0000016kOuYAI"
            },
            "FirstName" : null,
            "LastName" : "Wong",
            "Id" : "003RO0000016kOuYAI"
         },
         "httpHeaders" : { },
         "httpStatusCode" : 200,
         "referenceId" : "refContact"
      },
      {
         "body" : [
            {
               "errorCode" : "PROCESSING_HALTED",
              "message" : "Invalid reference specified. No value for refContact.FirstName
 found in refContact.
              Provided referenceId ('refContact.FirstName') must start with a letter or a
 number,
              and it can contain only letters, numbers and underscores ('_')."
            }
         ],
         "httpHeaders" : { },
         "httpStatusCode" : 400,
         "referenceId" : "newContact"
      }
   ]
}
```


That example assumes that `allOrNone` is set to false. If it's true, the whole composite request is rolled back. See allOrNone Parameters in Composite and Collections Requests.

Responses for Version 52.0 and Later

In API version 52.0 and later, the request succeeds.


```
{
   "compositeResponse" : [
      {
         "body" : {
```

```
            "attributes" : {
               "type" : "Contact",
               "url" : "/services/data/v51.0/sobjects/Contact/003RO0000016kOuYAI"
            },
            "FirstName" : null,
            "LastName" : "Wong",
            "Id" : "003RO0000016kOuYAI"
         },
         "httpHeaders" : { },
         "httpStatusCode" : 200,
         "referenceId" : "refContact"
      },
      {
         "body" : {
            "id" : "003RO0000016kRAYAY",
            "success" : true,
            "errors" : [ ]
         },
         "httpHeaders" : {
            "Location" : "/services/data/v51.0/sobjects/Contact/003RO0000016kRAYAY"
         },
         "httpStatusCode" : 201,
         "referenceId" : "newContact"
      }
   ]
}
```

Behavior for References to Fields That Aren't Specified in the Parent Request

In dependent subrequests, you must always only use fields that are explicitly selected in parent requests. If this practice isn't followed, the API's behavior depends on the API version. (The pertinent API version is that used to make the composite request. That version isn't necessarily the same as the API version specified in the `url` parameters in the subrequests.)

For example, consider the following request. It attempts to:

1. Locate a specific Contact.
2. Use `@{refContact.records[0].AccountId}` to get that Contact's Account ID.

However, the parent request doesn't explicitly query for the `AccountId`.

```
POST https://MyDomainName.my.salesforce.com/services/data/vXX.X/composite -H "Authorization:
 Bearer token"

{
   "compositeRequest" : [
      {
         "method" : "GET",
         "url" : "/services/data/v51.0/query?q=select Id, Account.Name from Contact where
 Id='003RO0000016kOuYAI'",
         "referenceId" : "refContact"
      },
      {
         "method" : "GET",
         "url" : "/services/data/v50.0/query?q=select Name from Account where Id =
```

```
'@{refContact.records[0].AccountId}'",
         "referenceId" : "refAccount"
      }
   ]
}
```

Responses for Version 51.0 and Earlier

In API version 51.0 and earlier, there are *rare cases* where such a request succeeds.

Note: The fact that a request like this sometimes succeeds should never be relied upon. If you plan to use a field from a parent request's result, you should always explicitly select that field in the parent request.

Responses for Version 52.0 and Later

In API version 52.0 and later, the request always fails:


```
{
   "compositeResponse" : [
      {
         "body" : {
            "totalSize" : 1,
            "done" : true,
            "records" : [
               {
                  "attributes" : {
                     "type" : "Contact",
                     "url" : "/services/data/v51.0/sobjects/Contact/003RO0000016kOuYAI"
                  },
                  "Id" : "003RO0000016kOuYAI",
                  "Account" : {
                     "attributes" : {
                        "type" : "Account",
                       "url" : "/services/data/v51.0/sobjects/Account/001RO000001nGbUYAU"

                     },
                     "Name" : "City Medical Center"
                  }
               }
            ]
         },
         "httpHeaders" : { },
         "httpStatusCode" : 200,
         "referenceId" : "refContact"
      },
      {
         "body" : [
            {
               "errorCode" : "PROCESSING_HALTED",
               "message" : "Invalid reference specified. No value for
refContact.records[0].AccountId found in refContact.
               Provided referenceId ('refContact.records[0].AccountId') must start with a
 letter or a number, and it can contain
               only letters, numbers and underscores ('_')."
            }
         ],
```

```
            "httpHeaders" : { },
            "httpStatusCode" : 400,
            "referenceId" : "refAccount"
         }
      ]
}
```

To make such a request work, you must explicitly obtain the `AccountId` in the parent request:

```
{
   "compositeRequest" : [
      {
         "method" : "GET",
         "url" : "/services/data/v51.0/query?q=select Id, Account.Name, AccountId from
Contact where Id='003RO0000016kOuYAI'",
         "referenceId" : "refContact"
      },
      {
         "method" : "GET",
         "url" : "/services/data/v50.0/query?q=select Name from Account where Id =
'@{refContact.records[0].AccountId}'",
         "referenceId" : "refAccount"
      }
   ]
}
```

Get a List of Composite Resources

Gets a list of URIs for other composite resources.

Syntax

URI

`/services/data/vXX.X/composite`

Formats

JSON

HTTP method

GET

Authentication

`Authorization: Bearer token`

Parameters

None required

Request body

None required

Example

Example Request

```
curl https://MyDomainName.my.salesforce.com/services/data/v60.0/composite/ -H
"Authorization: Bearer token"
```

Example Response Body

```
{
    "tree": "/services/data/v54.0/composite/tree",
    "batch": "/services/data/v54.0/composite/batch",
    "sobjects": "/services/data/v54.0/composite/sobjects",
    "graph": "/services/data/v54.0/composite/graph"
}
```

Composite Graph

The composite graph resource lets you submit composite graph operations. This resource is available in REST API version 50.0 and later.

Note: The response bodies and HTTP statuses of the requests are returned in a single response body. The entire request counts as a single call toward your API limits.

Syntax

URI

`/services/data/vXX.X/composite/graph`

Formats

JSON

HTTP methods

POST

Authentication

`Authorization: Bearer token`

Request parameters

None

Request Body

```
{
    "graphId" : "graphId",
    "compositeRequest" : [
        compositeSubrequest,
        compositeSubrequest,
        ...
    ]
}
```

where each ***compositeSubrequest*** is a composite subrequest.

Response Body

```
{
    "graphs" : [
        {
            "graphId" : "graphId",
            "graphResponse" : {
                "compositeResponse" : [
                    compositeSubrequestResult,
                    compositeSubrequestResult,
                    compositeSubrequestResult,
                    ...
                ]
            },
            "isSuccessful" : flag
        },
        ...
    ]
}
```

Name	Type	Description
`graphs`	Array of graph responses.	
`graphId`	String	The identifier of the graph.
`graphResponse`	Object	The response of the request.
`compositeResponse`	Array of composite subrequest results on page 366.	Results for each node in the graph.
`isSuccessful`	Boolean	Whether this graph was processed successfully (`true`) or not (`false`).

Example

Example Request

```
curl -X POST https://MyDomainName.my.salesforce.com/services/data/v60.0/composite/graph
 -H "Authorization: Bearer token" -H "Content-Type: application/json" -d
"@graphRequestBody.json"
```

Example Request Body

```
{
    "graphs" : [
        {
            "graphId" : "1",
            "compositeRequest" : [
                {
                    "url" : "/services/data/v60.0/sobjects/Account/",
                    "body" : {
                        "name" : "Cloudy Consulting"
                    },
```

```
                "method" : "POST",
                "referenceId" : "reference_id_account_1"
            },
            {
                "url" : "/services/data/v60.0/sobjects/Contact/",
                "body" : {
                    "FirstName" : "Nellie",
                    "LastName" : "Cashman",
                    "AccountId" : "@{reference_id_account_1.id}"
                },
                "method" : "POST",
                "referenceId" : "reference_id_contact_1"
            },
            {
                "url" : "/services/data/v60.0/sobjects/Opportunity/",
                "body" : {
                    "CloseDate" : "2024-05-22",
                    "StageName" : "Prospecting",
                    "Name" : "Opportunity 1",
                    "AccountId" : "@{reference_id_account_1.id}"
                },
                "method" : "POST",
                "referenceId" : "reference_id_opportunity_1"
            }
        ]
    },
    {
        "graphId" : "2",
        "compositeRequest" : [
            {
                "url" : "/services/data/v60.0/sobjects/Account/",
                "body" : {
                    "name" : "Easy Spaces"
                },
                "method" : "POST",
                "referenceId" : "reference_id_account_2"
            },
            {
                "url" : "/services/data/v60.0/sobjects/Contact/",
                "body" : {
                    "FirstName" : "Charlie",
                    "LastName" : "Dawson",
                    "AccountId" : "@{reference_id_account_2.id}"
                },
                "method" : "POST",
                "referenceId" : "reference_id_contact_2"
            }
        ]
    }
  ]
}
```

Example Response Body


```
{
   "graphs" : [
      {
         "graphId" : "1",
         "graphResponse" : {
            "compositeResponse" : [
               {
                  "body" : {
                     "id" : "001R00000064wc7IAA",
                     "success" : true,
                     "errors" : [ ]
                  },
                  "httpHeaders" : {
                     "Location" :
"/services/data/v60.0/sobjects/Account/001R00000064wc7IAA"
                  },
                  "httpStatusCode" : 201,
                  "referenceId" : "reference_id_account_1"
               },
               {
                  "body" : {
                     "id" : "003R000000DDMlTIAX",
                     "success" : true,
                     "errors" : [ ]
                  },
                  "httpHeaders" : {
                     "Location" :
"/services/data/v60.0/sobjects/Contact/003R000000DDMlTIAX"
                  },
                  "httpStatusCode" : 201,
                  "referenceId" : "reference_id_contact_1"
               },
               {
                  "body" : {
                     "id" : "006R0000003FPYxIAO",
                     "success" : true,
                     "errors" : [ ]
                  },
                  "httpHeaders" : {
                     "Location" :
"/services/data/v60.0/sobjects/Opportunity/006R0000003FPYxIAO"
                  },
                  "httpStatusCode" : 201,
                  "referenceId" : "reference_id_opportunity_1"
               }
            ]
         },
         "isSuccessful" : true
      },
      {
         "graphId" : "2",
         "graphResponse" : {
```

```
            "compositeResponse" : [
               {
                  "body" : {
                     "id" : "001R00000064wc8IAA",
                     "success" : true,
                     "errors" : [ ]
                  },
                  "httpHeaders" : {
                     "Location" :
"/services/data/v60.0/sobjects/Account/001R00000064wc8IAA"
                  },
                  "httpStatusCode" : 201,
                  "referenceId" : "reference_id_account_2"
               },
               {
                  "body" : {
                     "id" : "003R000000DDMlUIAX",
                     "success" : true,
                     "errors" : [ ]
                  },
                  "httpHeaders" : {
                     "Location" :
"/services/data/v60.0/sobjects/Contact/003R000000DDMlUIAX"
                  },
                  "httpStatusCode" : 201,
                  "referenceId" : "reference_id_contact_2"
               }
            ]
         },
         "isSuccessful" : true
      }
   ]
}
```


Composite Subrequest

The composite subrequest describes a subrequest to execute with the Composite Graph resource.

Properties

Name	Type	Description
`body`	Varies	Optional. The input body for the subrequest. The contents depend on the request specified in the `url` property. Referenceable types are DateTime, String, Boolean, Byte, Character, Short, Integer, Long, Double, and Float.
`httpHeaders`	Map<String, String>	Optional.

Name	Type	Description
		Request headers and their values to include with the subrequest. You can include any header supported by the requested resource except for the following three headers. • Accept • Authorization • Content-Type Subrequests inherit these three header values from the top-level request. Don't specify these headers in a subrequest. If you do, the top-level request fails and returns an HTTP 400 response. Subrequests that contain valid HTTP headers can't be collated, which slows the processing speed of the request.
`method`	String	Required. The method to use with the requested resource. Possible values are DELETE, GET, PATCH, and POST (case-sensitive). For a list of valid methods, refer to the documentation for the requested resource. This method is distinct from the POST method that is used to submit the composite graph request. The method specified here is the one that operates (within the graph) on the resource specified in the `url`. • If the `url` is `/services/data/vXX.X/sobject/`***`sObject`*** then POST is supported. (See sObject Basic Information.) • If the `url` is `/services/data/v`***`XX.X`***`/sobject/`***`sObject`***`/`***`id`*** then DELETE, GET, and PATCH are supported. (See sObject Rows.) • If the `url` is `/services/data/v`***`XX.X`***`/sobject/`***`sObject`***`/`***`customFieldName`***`/`***`externalId`*** then DELETE, GET, PATCH, and POST are supported. You can use PATCH to upsert via an external ID). See Insert or Update (Upsert) a Record Using an External ID.
`referenceId`	String	Required. Reference ID that maps to the subrequest's response and can be used to reference the response in later subrequests. You can reference the `referenceId` in either the body or URL of a subrequest. Use this syntax to include a reference: `@{referenceId.FieldName}`. You can use two operators with the reference ID. The `.` operator references a field on a JSON object in the response. For example, say you retrieve an account record's data in one subrequest and assign the reference ID `Account1Data` to the output. You can refer to the account's name in later subrequests like this: `@{Account1Data.Name}`. The `[]` operator indexes a JSON collection in the response. For example, say you request basic account information with the sObject Basic Information

Name	Type	Description
		resource in one subrequest and assign the reference ID `AccountInfo` to the output. Part of the response includes a collection of recently created accounts. You can refer to the ID of the first account in the recent items collection like this: `@{AccountInfo.recentItems[0].Id}`. You can use each operator recursively as long as it makes sense in the context of the response. For example, to refer to the billing city component of an account's compound address field: `@{NewAccount.BillingAddress.city}`. `referenceId` is case-sensitive, so pay close attention to the case of the fields you're referring to. See Usage. • The `referenceId` must start with a letter or a number. • The `referenceId` must not contain anything besides letters, numbers, or underscores ('_').

`url`	String	Required.
		The resource to request.
		• The URL can include any query string parameters that the subrequest supports. The query string must be URL-encoded.
		• You can use parameters to filter response bodies.
		• Only the following URLs are supported:
		  – `/services/data/vXX.X/sobject/sObject`
		  – `/services/data/vXX.X/sobject/sObject/id`
		  – `/services/data/vXX.X/sobject/sObject/customFieldName/externalId`
		• The version number ***XX.X*** must be 50.0 or later.

Examples

Example 1

```
{
   "body" : {
     "Name" : "Sample Account"
   },
   "method" : "POST",
   "referenceId" : "refAccount",
   "url" : "/services/data/v60.0/sobjects/Account"
}
```

Example 2

```
{
   "method" : "GET",
   "referenceId" : "NewAccountFields",
```

```
    "url" : "/services/data/v60.0/sobjects/Account/@{refAccount.id}"
}
```

Usage

Because `referenceId` is case-sensitive, it's important to note the case of the fields that you're referring to. The same field can use different cases in different contexts. For example, when you create a record, the ID field appears as `id` in the response. But when you access a record's data with the sObject Rows resource, the ID field appears as `Id`. In the Example 2, the `@{refAccount.id}` reference is valid because `refAccount` refers to the response from a POST like that shown in Example 1. If you use `Id` instead (mixed case rather than all lowercase), as in `@{refAccount.Id}`, you get an error when sending the request because the reference ID uses the wrong case.

If one request reaches the maximum number of graph failures, processing stops for any remaining graphs. The response includes the `PROCESSING_HALTED` error and information about any failed graphs. See Composite Graph Limits for the current limits.

Results

Results for subrequests are the same as that for other Composite requests. See Composite Subrequest Result on page 366.

Composite Graph Limits

These limits are specific to composite graph resources. Review the platform API limits and allocations for a comprehensive list of all applicable limits to composite graph API resources.

General Limits on Graphs

Item	Limit
Maximum number of graphs in one payload.	75
Maximum depth of a graph.	15
Maximum number of nodes used in one graph.	500
Maximum number of different nodes in one payload. Nodes are considered different if they use resources from different API versions or different types of objects. For example: • `/services/data/v`**`50.0`**`/sobjects/account` and `/services/data/v`**`52.0`**`/sobjects/account` are considered different. • `/services/data/v`*`XX.X`*`/sobjects/`**`account`** and `/services/data/v`*`XX.X`*`/sobjects/`**`contact`** are considered different.	15
Maximum number of graph failures within one request.	14

Limits on Nodes

Item	Limit
Maximum number of nodes supported in one payload.	500 (For example, there can be one graph with 500 nodes, or 50 graphs with 10 nodes each.)

SEE ALSO:

API Request Limits and Allocations

Composite Batch

Executes up to 25 subrequests in a single request. The response bodies and HTTP statuses of the subrequests in the batch are returned in a single response body. Each subrequest counts against rate limits.

The requests in a batch are called subrequests. All subrequests are executed in the context of the same user. Subrequests are independent, and you can't pass information between them. Subrequests execute serially in their order in the request body. When a subrequest executes successfully, it commits its data. Commits are reflected in the output of later subrequests. If a subrequest fails, commits made by previous subrequests aren't rolled back. If a batch request doesn't complete within 10 minutes, the batch times out and the remaining subrequests aren't executed.

Batching for the following resources and resource groups is available in API version 34.0 and later.

- Limits—`/services/data/vXX.X/limits`
- sObject resources (except sObject Blob Retrieve and sObject Rich Text Image Retrieve)—`/services/data/vXX.X/sobjects/`
- Query—`/services/data/vXX.X/query/?q=soql`
- QueryAll—`/services/data/vXX.X/queryAll/?q=soql`
- Search—`/services/data/vXX.X/search/?q=sosl`
- Connect resources—`/services/data/vXX.X/connect/`
- Chatter resources—`/services/data/vXX.X/chatter/`

Batching for the following resources and resource groups is available in API version 35.0 and later.

- Actions—`vXX.X/actions`

The API version of the resource accessed in each subrequest must be no earlier than 34.0 and no later than the Batch version in the top-level request. For example, if you post a Batch request to `/services/data/v35.0/composite/batch`, you can include subrequests that access version 34.0 or 35.0 resources. But if you post a Batch request to `/services/data/v34.0/composite/batch`, you can only include subrequests that access version 34.0 resources.

Syntax

URI

`/services/data/vXX.X/composite/batch`

Formats

JSON, XML

HTTP method

POST

Authentication

`Authorization: Bearer token`

Parameters

None required

Request body

Batch Request Body on page 384

Response body

Batch Response Body on page 386

Example

For an example of using the Composite Batch resource, see Update a Record and Get Its Field Values in a Single Request on page 101.

Batch Request Body

Describes a collection of subrequests to execute with the Composite Batch resource.

Batch Collection Input

The request body contains a `batchRequests` collection that includes subrequests to execute.

Properties

Name	Type	Description	Required or Optional
`batchRequests`	Subrequest[]	Collection of subrequests to execute.	Required
`haltOnError`	Boolean	Controls whether a batch continues to process after a subrequest fails. The default is `false`. If the value is `false` and a subrequest in the batch doesn't complete, Salesforce attempts to execute the subsequent subrequests in the batch. If the value is `true` and a subrequest in the batch doesn't complete due to an HTTP response in the 400 or 500 range, Salesforce halts execution. It returns an HTTP 412 status code and a `BATCH_PROCESSING_HALTED` error message for each subsequent subrequest. The top-level request to `/composite/batch` returns HTTP 200, and the `hasErrors` property in the response is set to `true`. This setting is only applied during subrequest processing, and not during initial request deserialization. If an error is detected during deserialization, such as a syntax error in the Subrequest request data, Salesforce returns an HTTP 400 `Bad Request` error without processing any subrequests, regardless of the value	Optional

Name	Type	Description	Required or Optional
		of `haltOnError`. An example where this could occur is if a subrequest has an invalid `method` or `url` field.	

Root XML Tag

`<batch>`

JSON example

```
{
"batchRequests" : [
    {
    "method" : "PATCH",
    "url" : "v60.0/sobjects/account/001D000000K0fXOIAZ",
    "richInput" : {"Name" : "NewName"}
    },{
    "method" : "GET",
    "url" : "v60.0/sobjects/account/001D000000K0fXOIAZ?fields=Name,BillingPostalCode"
    }]
}
```

Subrequest

Contains the resource, method, and accompanying information for the subrequest.

Properties

Name	Type	Description	Required or Optional
`binaryPartName`	String	The name of the binary part in the multipart request. When multiple binary parts are uploaded in one batch request, this value is used to map a request to its binary part. To prevent name collisions, use a unique value for each `binaryPartName` property in a batch request. If this value exists, a `binaryPartNameAlias` value must also exist.	Optional
`binaryPartNameAlias`	String	The `name` parameter in the Content-Disposition header of the binary body part. Different resources expect different values. See Insert or Update Blob Data. If this value exists, a `binaryPartName` value must also exist.	Optional
`method`	String	The method to use with the requested resource. For a list of valid methods, refer to the documentation for the requested resource.	Required
`richInput`		The input body for the request. The type depends on the request specified in the `url` property.	Optional

Name	Type	Description	Required or Optional
`url`	String	The resource to request. • The URL can include any query string parameters that the subrequest supports. The query string must be URL-encoded. • You can use parameters to filter response bodies. • You cannot apply headers at the subrequest level.	Required

Root XML Tag

`<request>`

JSON example

```
{
    "method" : "GET",
    "url" : "v60.0/sobjects/account/001D000000K0fXOIAZ?fields=Name,BillingPostalCode"
}
```

Batch Response Body

Describes the result of a Composite Batch request.

Batch Results

Properties

Name	Type	Description
`hasErrors`	Boolean	`true` if at least one of the results in the result set is an HTTP status code in the 400 or 500 range; `false` otherwise.
`results`	Subrequest Result[]	Collection of subrequest results.

JSON example

```
{
   "hasErrors" : false,
   "results" : [{
      "statusCode" : 204,
      "result" : null
      },{
      "statusCode" : 200,
      "result": {
         "attributes" : {
            "type" : "Account",
            "url" : "/services/data/v60.0/sobjects/Account/001D000000K0fXOIAZ"
         },
         "Name" : "NewName",
```

```
            "BillingPostalCode" : "94105",
            "Id" : "001D000000K0fXOIAZ"
        }
    }]
}
```

Subrequest Result

Properties

Name	Type	Description
`result`	The type depends on the response type of the subrequest. **Important:** If the result is an error, the type is a collection containing the error message and error code.	The response body of this subrequest.
`statusCode`	Integer	An HTTP status code indicating the status of this subrequest.

JSON example

```
{
    "attributes" : {
        "type" : "Account",
        "url" : "/services/data/v60.0/sobjects/Account/001D000000K0fXOIAZ"
    },
    "Name" : "NewName",
    "BillingPostalCode" : "94015",
    "Id" : "001D000000K0fXOIAZ"
}
```

sObject Tree

Creates one or more sObject trees with root records of the specified type. An sObject tree is a collection of nested, parent-child records with a single root record.

In the request data, you supply the record hierarchies, required and optional field values, each record's type, and a reference ID for each record. Objects are created in the same order that they're listed in the request. Upon success, the response contains the IDs of the created records. If an error occurs while creating a record, the entire request fails. In this case, the response contains only the reference ID of the record that caused the error and the error information. The response bodies and HTTP statuses of the requests are returned in a single response body. The entire request counts as a single call toward your API limits.

The request can contain the following:

- Up to a total of 200 records across all trees
- Up to five records of different types
- sObject trees up to five levels deep

Because an sObject tree can contain a single record, you can use this resource to create up to 200 unrelated records of the same type.

When the request is processed and records are created, triggers, processes, and workflow rules fire separately for each of the following groups of records.

- Root records across all sObject trees in the request
- All second-level records of the same type—for example, second-level Contacts across all sObject trees in the request
- All third-level records of the same type
- All fourth-level records of the same type
- All fifth-level records of the same type

Syntax

URI

`/services/data/vXX.X/composite/tree/sObjectName`

Formats

JSON, XML

HTTP method

POST

Authentication

`Authorization: Bearer token`

Parameters

None required

Request body

sObject Tree Request Body on page 388

Response body

sObject Tree Response Body on page 391

Example

- For an example of creating unrelated records of the same type, see Create Multiple Records on page 105.
- For an example of creating nested records, see Create Nested Records on page 103.

sObject Tree Request Body

Describes a collection of sObject trees to create with the sObject Tree resource.

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.

sObject Tree Collection Input

The request body contains a `records` collection that includes sObject trees.

Properties

Name	Type	Description	Required or Optional
`records`	sObject Tree Input[]	Collection of record trees to create. Each tree's root record type must correspond to the sObject specified in the sObject Tree URI.	Required

Root XML Tag

<SObjectTreeRequest>

JSON example

```
{
"records" :[{
    "attributes" : {"type" : "Account", "referenceId" : "ref1"},
    "name" : "SampleAccount",
    "phone" : "1234567890",
    "website" : "www.salesforce.com",
    "numberOfEmployees" : "100",
    "industry" : "Banking",
    "Contacts" : {
      "records" : [{
         "attributes" : {"type" : "Contact", "referenceId" : "ref2"},
         "lastname" : "Smith",
         "title" : "President",
         "email" : "sample@salesforce.com"
         },{
         "attributes" : {"type" : "Contact", "referenceId" : "ref3"},
         "lastname" : "Evans",
         "title" : "Vice President",
         "email" : "sample@salesforce.com"
         }]
      }
    },{
    "attributes" : {"type" : "Account", "referenceId" : "ref4"},
    "name" : "SampleAccount2",
    "phone" : "1234567890",
    "website" : "www.salesforce2.com",
    "numberOfEmployees" : "100",
    "industry" : "Banking"
     }]
}
```

XML example

```
<SObjectTreeRequest>
    <records type="Account" referenceId="ref1">
        <name>SampleAccount</name>
        <phone>1234567890</phone>
        <website>www.salesforce.com</website>
        <numberOfEmployees>100</numberOfEmployees>
        <industry>Banking</industry>
```

```
        <Contacts>
            <records type="Contact" referenceId="ref2">
                <lastname>Smith</lastname>
                <title>President</title>
                <email>sample@salesforce.com</email>
            </records>
            <records type="Contact" referenceId="ref3">
                <lastname>Evans</lastname>
                <title>Vice President</title>
                <email>sample@salesforce.com</email>
            </records>
        </Contacts>
    </records>
    <records type="Account" referenceId="ref4">
        <name>SampleAccount2</name>
        <phone>1234567890</phone>
        <website>www.salesforce2.com</website>
        <numberOfEmployees>100</numberOfEmployees>
        <industry>Banking</industry>
    </records>
</SObjectTreeRequest>
```

sObject Tree Input

An sObject tree is a recursive data structure that contains a root record, its data, and its child records represented as other sObject trees.

Properties

Name	Type	Description	Required or Optional
`attributes`	Collection	Attributes for this record. The `attributes` property contains two subproperties: • `type` (required)—This record's type. • `referenceId` (required)—Reference ID for this record. Must be unique in the context of the request and start with an alphanumeric character. In XML, include attributes inside the `records` element.	Required
Required object fields	Depends on field	Required fields and field values for this record.	Required
Optional object fields	Depends on field	Optional fields and field values for this record.	Optional
Child relationships	sObject Tree Collection Input	This record's child relationships, such as an account's child contacts. Child relationships are either master-detail or lookup relationships. To view an object's valid child relationships, use the sObject Describe resource or Schema Builder. The value of this property is an sObject Tree Collection Input that contains child sObject trees.	Optional

Root XML Tag

<records>

JSON example

```
{
"attributes" : {"type" : "Account", "referenceId" : "ref1"},
"name" : "SampleAccount",
"phone" : "1234567890",
"website" : "www.salesforce.com",
"NumberOfEmployees" : "100",
"industry" : "Banking",
"Contacts" : {
  "records" : [{
     "attributes" : {"type" : "Contact", "referenceId" : "ref2"},
     "lastname" : "Smith",
     "title" : "President",
     "email" : "sample@salesforce.com"
      },{
     "attributes" : {"type" : "Contact", "referenceId" : "ref3"},
     "lastname" : "Evans",
     "title" : "Vice President",
     "email" : "sample@salesforce.com"
     }]
  }
}
```

XML example

```
<records type="Account" referenceId="ref1">
    <name>SampleAccount</name>
    <phone>1234567890</phone>
    <website>www.salesforce.com</website>
    <numberOfEmployees>100</numberOfEmployees>
    <industry>Banking</industry>
    <Contacts>
        <records type="Contact" referenceId="ref2">
            <lastname>Smith</lastname>
            <title>President</title>
            <email>sample@salesforce.com</email>
        </records>
        <records type="Contact" referenceId="ref3">
            <lastname>Evans</lastname>
            <title>Vice President</title>
            <email>sample@salesforce.com</email>
        </records>
    </Contacts>
</records>
```

sObject Tree Response Body

Describes the result of an sObject Tree request.

Properties

Name	Type	Description
`hasErrors`	Boolean	`true` if an error occurred while creating a record; `false` otherwise.
`results`	Collection	Upon success, `results` contains the reference ID of each requested record and its new record ID. Upon failure, it contains only the reference ID of the record that caused the error, error status code, error message, and fields related to the error. In the case of duplicate reference IDs, `results` contains one item for each instance of the duplicate ID.

JSON example upon success

```
{
    "hasErrors" : false,
    "results" : [{
     "referenceId" : "ref1",
     "id" : "001D000000K0fXOIAZ"
     },{
     "referenceId" : "ref4",
     "id" : "001D000000K0fXPIAZ"
     },{
     "referenceId" : "ref2",
     "id" : "003D000000QV9n2IAD"
     },{
     "referenceId" : "ref3",
     "id" : "003D000000QV9n3IAD"
     }]
}
```

XML example upon success

```
<?xml version="1.0" encoding="UTF-8"?>
<SObjectTreeResponse>
    <hasErrors>false</hasErrors>
    <results>
        <id>001D000000K0fXOIAZ</id>
        <referenceId>ref1</referenceId>
    </results>
    <results>
        <id>001D000000K0fXPIAZ</id>
        <referenceId>ref4</referenceId>
    </results>
    <results>
        <id>003D000000QV9n2IAD</id>
        <referenceId>ref2</referenceId>
    </results>
    <results>
        <id>003D000000QV9n3IAD</id>
        <referenceId>ref3</referenceId>
    </results>
</SObjectTreeResponse>
```

JSON example upon failure


```
{
   "hasErrors" : true,
   "results" : [{
     "referenceId" : "ref2",
     "errors" : [{
       "statusCode" : "INVALID_EMAIL_ADDRESS",
       "message" : "Email: invalid email address: 123",
       "fields" : [ "Email" ]
       }]
     }]
}
```


XML example upon failure


```
<SObjectTreeResponse>
    <hasErrors>true</hasErrors>
    <results>
        <errors>
            <fields>Email</fields>
            <message>Email: invalid email address: 123</message>
            <statusCode>INVALID_EMAIL_ADDRESS</statusCode>
        </errors>
        <referenceId>ref2</referenceId>
    </results>
</SObjectTreeResponse>
```


sObject Collections

Executes actions on multiple records in one request. Use sObject Collections to reduce the number of round-trips between the client and server. The response bodies and HTTP statuses of the requests are returned in a single response body. The entire request counts as a single call toward your API limits. This resource is available in API version 42.0 and later.

The parameters, request body, and response body of an sObject Collections request depend on the HTTP method. For details, see the specific operation.

Create Records Using sObject Collections

Use a POST request with sObject Collections to add up to 200 records, returning a list of SaveResult objects. You can choose whether to roll back the entire request when an error occurs.

- The list can contain up to 200 objects.
- The list can contain objects of different types, including custom objects.
- Each object must contain an attributes map. The map must contain a value for `type`.

  Note: Using sObject Collections to insert blob data requires more values in the attributes map. For more information, see Using sObject Collections to Insert a Collection of Blob Records.

- Objects are created in the order they're listed. The SaveResult objects are returned in the order in which the create requests were specified.

- If the request body includes objects of more than one type, they are processed as chunks. For example, if the incoming objects are `{account1, account2, contact1, account3}`, the request is processed in three chunks: `{{account1, account2}, {contact1}, {account3}}`. A single request can process up to 10 chunks.
- You can't create records for multiple object types in one call when one of the types is related to a feature in the Salesforce Setup area.

If the request isn't well formed, the API returns a `400 Bad Request` HTTP Status. Fix the syntax of the request and try again. If the request is well formed, the API returns a `200 OK` HTTP Status. If an item was processed successfully, the `success` flag shows for that item. Error information is returned in the `errors` array.

Syntax

URI

`/services/data/vXX.X/composite/sobjects`

Formats

JSON, XML

HTTP Method

POST

Authentication

`Authorization: Bearer token`

Parameters

Parameter	Description
`allOrNone`	Optional. Indicates whether to roll back the entire request when the creation of any object fails (`true`) or to continue with the independent creation of other objects in the request. The default is `false`. Note: If the sObject Collections request is embedded in a Composite request, the Composite request's `allOrNone` parameter can also affect the results. See allOrNone Parameters in Composite and Collections Requests.
`records`	Required. A list of sObjects. In a POST request using sObject Collections, set the `type` attribute for each object, but don't set the `id` field for any object.

Example

Example Request

```
curl -X POST
https://MyDomainName.my.salesforce.com/services/data/v60.0/composite/sobjects/ -H
"Authorization: Bearer token" -H "Content-Type: application/json" -d
"@exampleRequestBody.json"
```

Example Request Body

```
{
   "allOrNone" : false,
   "records" : [{
      "attributes" : {"type" : "Account"},
```

```
      "Name" : "example.com",
      "BillingCity" : "San Francisco"
   }, {
      "attributes" : {"type" : "Contact"},
      "LastName" : "Johnson",
      "FirstName" : "Erica"
   }]
}
```

Example Response Body

```
HTTP/1.1 200 OK

[
   {
      "id" : "001RM000003oLnnYAE",
      "success" : true,
      "errors" : [ ]
   },
   {
      "id" : "003RM0000068xV6YAI",
      "success" : true,
      "errors" : [ ]
   }
]
```

Example Response Body (Some Items Failed and `allOrNone` is `false`)

```
HTTP/1.1 200 OK

[
   {
      "success" : false,
      "errors" : [
         {
            "statusCode" : "DUPLICATES_DETECTED",
            "message" : "Use one of these records?",
            "fields" : [ ]
         }
      ]
   },
   {
      "id" : "003RM0000068xVCYAY",
      "success" : true,
      "errors" : [ ]
   }
]
```

Example Response Body (Some Items Failed and `allOrNone` is `true`)

```
HTTP/1.1 200 OK

[
   {
      "success" : false,
      "errors" : [
```

```
            {
               "statusCode" : "DUPLICATES_DETECTED",
               "message" : "Use one of these records?",
               "fields" : [ ]
            }
         ]
      },
      {
         "success" : false,
         "errors" : [
            {
               "statusCode" : "ALL_OR_NONE_OPERATION_ROLLED_BACK",
               "message" : "Record rolled back because not all records were valid and the
 request was using AllOrNone header",
               "fields" : [ ]
            }
         ]
      }
]
```


Get Records Using sObject Collections

Use a GET request with sObject Collections to get one or more records of the same object type. A list of sObjects that represents the individual records of the specified type is returned. The number of sObjects returned matches the number of IDs passed in the request.

You can specify approximately 800 IDs before the URL length causes the HTTP 414 error `URI too long`.

Note: The sObject Blob Retrieve on page 156 resource isn't compatible with Composite API requests, because it returns binary data instead of data in JSON or XML formats. Instead, make individual sObject Blob Retrieve requests to retrieve blob data.

- If you specify an invalid field name or a field name that you don't have permission to read, `HTTP 400 Bad Request` is returned.
- If you don't have access to an object, or if a passed ID is invalid, the array returns null for that object.

Syntax

URI

`/services/data/v`***`XX.X`***`/composite/sobjects/`***`sObject`***

Formats

JSON, XML

HTTP Method

GET

Authentication

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

Parameters

Parameter	Description
`recordIds`	Required. A list of one or more IDs of the objects to return. All IDs must belong to the same object type.

Parameter	Description
`fieldNames`	Required. A list of fields to include in the response. The field names you specify must be valid, and you must have read-level permissions to each field.

Example

Example Request

```
curl
https://MyDomainName.my.salesforce.com/services/data/v60.0/composite/sobjects/Account?ids=001xx000003DGb1AAG,001xx000003DGb0AAG,001xx000003DGb9AAG&fields=id,name
 -H "Authorization: Bearer token"
```

Example Response Body

```
[
   {
      "attributes" : {
         "type" : "Account",
         "url" : "/services/data/v60.0/sobjects/Account/001xx000003DGb1AAG"
      },
      "Id" : "001xx000003DGb1AAG",
      "Name" : "Acme"
   },
   {
      "attributes" : {
         "type" : "Account",
         "url" : "/services/data/v60.0/sobjects/Account/001xx000003DGb0AAG"
      },
      "Id" : "001xx000003DGb0AAG",
      "Name" : "Global Media"
   },
   null
]
```

Get Records With a Request Body Using sObject Collections

Use a POST request with sObject Collections to get one or more records of the same object type. A list of sObjects that represents the individual records of the specified type is returned. The number of sObjects returned matches the number of IDs passed in the request.

The request retrieves up to 2,000 records of the same object type

- If you specify an invalid field name or a field name that you don't have permission to read, `HTTP 400 Bad Request` is returned.
- If you don't have access to an object, or if a passed ID is invalid, the array returns null for that object.

Note: The sObject Blob Retrieve on page 156 resource isn't compatible with Composite API requests, because it returns binary data instead of data in JSON or XML formats. Instead, make individual sObject Blob Retrieve requests to retrieve blob data.

Syntax

URI

`/services/data/vXX.X/composite/sobjects/sObject`

Formats

JSON, XML

HTTP Method

POST

Authentication

`Authorization: Bearer token`

Request Body

```
{
   "ids" : ["recordIds"],
   "fields" : ["fieldName"]
}
```

Parameters

Parameter	Description
`recordIds`	Required. A list of one or more IDs of the objects to return. All IDs must belong to the same object type.
`fieldNames`	Required. A list of fields to include in the response. The field names you specify must be valid, and you must have read-level permissions to each field.

Example

Example Request

```
curl -X POST
https://MyDomainName.my.salesforce.com/services/data/v60.0/composite/sobjects/Account
-H "Authorization: Bearer token" -H "Content-Type: application/json" -d
"@exampleRequestBody.json"
```

Example Request Body

```
{
   "ids" : ["001xx000003DGb1AAG", "001xx000003DGb0AAG", "001xx000003DGb9AAG"],
   "fields" : ["id", "name"]
}
```

Example Response Body

```
[
   {
      "attributes" : {
         "type" : "Account",
         "url" : "/services/data/v60.0/sobjects/Account/001xx000003DGb1AAG"
      },
      "Id" : "001xx000003DGb1AAG",
      "Name" : "Acme"
   },
   {
      "attributes" : {
         "type" : "Account",
```

```
            "url" : "/services/data/v60.0/sobjects/Account/001xx000003DGb0AAG"
        },
        "Id" : "001xx000003DGb0AAG",
        "Name" : "Global Media"
    },
    null
]
```

Update Records Using sObject Collections

Use a PATCH request with sObject Collections to update up to 200 records, returning a list of SaveResult objects. You can choose whether to roll back the entire request when an error occurs.

- The list can contain up to 200 objects.
- The list can contain objects of different types, including custom objects.
- Each object must contain an attributes map. The map must contain a value for `type`.

> Note: Using sObject Collections to update blob data requires more values in the attributes map. For more information, see Using sObject Collections to Insert a Collection of Blob Records.

- Each object must contain an `id` field with a valid ID value.
- Objects are updated in the order they're listed. The SaveResult objects are returned in the order in which the update requests were specified.
- If the request body includes objects of more than one type, they are processed as chunks. For example, if the incoming objects are `{account1, account2, contact1, account3}`, the request is processed in three chunks: `{{account1, account2}, {contact1}, {account3}}`. A single request can process up to 10 chunks.
- You can't update records for multiple object types in one call when one of those types is related to a feature in the Salesforce Setup area.

If the request isn't well formed, the API returns a `400 Bad Request` HTTP Status. Fix the syntax of the request and try again. If the request is well formed, the API returns a `200 OK` HTTP Status. If an item was processed successfully, the `success` flag shows for that item. Error information is returned in the `errors` array.

Syntax

URI

`/services/data/v`***`XX.X`***`/composite/sobjects/`

Formats

JSON, XML

HTTP Method

PATCH

Authentication

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

Parameters

Parameter	Description
`allOrNone`	Optional. Indicates whether to roll back the entire request when the update of any object fails (`true`) or to continue with the independent update of other objects in the request. The default is `false`. Note: If the sObject Collections request is embedded in a Composite request, the Composite request's `allOrNone` parameter can also affect the results. See allOrNone Parameters in Composite and Collections Requests.
`records`	Required. A list of records. Set the `id` and `type` attribute for each record.

Example

Example Request

```
curl -X PATCH
https://MyDomainName.my.salesforce.com/services/data/v60.0/composite/sobjects/ -H
"Authorization: Bearer token" -H "Content-Type: application/json" -d
"@exampleRequestBody.json"
```

Example Request Body

```
{
   "allOrNone" : false,
   "records" : [{
      "attributes" : {"type" : "Account"},
      "id" : "001xx000003DGb2AAG",
      "NumberOfEmployees" : 27000
   },{
      "attributes" : {"type" : "Contact"},
      "id" : "003xx000004TmiQAAS",
      "Title" : "Lead Engineer"
   }]
}
```

Example Response Body

```
HTTP/1.1 200 OK

[
   {
      "id" : "001RM000003oCprYAE",
      "success" : true,
      "errors" : [ ]
   },
   {
      "id" : "003RM0000068og4YAA",
      "success" : true,
      "errors" : [ ]
   }
]
```

Example Response Body (Some Items Failed and `allOrNone` is `false`)

```
HTTP/1.1 200 OK

[
   {
      "id" : "001RM000003oCprYAE",
      "success" : true,
      "errors" : [ ]
   },
   {
      "success" : false,
      "errors" : [
         {
            "statusCode" : "MALFORMED_ID",
            "message" : "Contact ID: id value of incorrect type: 001xx000003DGb2999",
            "fields" : [
               "Id"
            ]
         }
      ]
   }
]
```

Example Response Body (Some Items Failed and `allOrNone` is `true`)

```
HTTP/1.1 200 OK

[
   {
      "id" : "001RM000003oCprYAE",
      "success" : false,
      "errors" : [
         {
            "statusCode" : "ALL_OR_NONE_OPERATION_ROLLED_BACK",
            "message" : "Record rolled back because not all records were valid and the
 request was using AllOrNone header",
            "fields" : [ ]
         }
      ]
   },
   {
      "success" : false,
      "errors" : [
         {
            "statusCode" : "MALFORMED_ID",
            "message" : "Contact ID: id value of incorrect type: 001xx000003DGb2999",
            "fields" : [
               "Id"
            ]
         }
      ]
   }
]
```

Upsert Records Using sObject Collections

Use a PATCH request with sObject Collections to either create or update (upsert) up to 200 records based on an external ID field. This method returns a list of UpsertResult objects. You can choose whether to roll back the entire request when an error occurs. This request is available in API version 46 and later.

- The list can contain up to 200 objects.
- The list can contain objects only of the type indicated in the request URI.
- Each object in the request body must contain an attributes map. The map must contain a value for `type`.

  Note: Using sObject Collections to insert blob data requires more values in the attributes map. For more information, see Using sObject Collections to Insert a Collection of Blob Records.

- Objects are created or updated in the order they're listed in the request body. The UpsertResult objects are returned in the same order.
- Only external ids are supported. Don't use record ids.

If the request isn't well formed, the API returns a `400 Bad Request` HTTP Status. Fix the syntax of the request and try again. If the request is well formed, the API returns a `200 OK` HTTP Status. If an item was processed successfully, the `success` flag shows for that item. Error information is returned in the `errors` array.

Syntax

URI

`/services/data/vXX.X/composite/sobjects/SobjectName/ExternalIdFieldName`

Formats

JSON, XML

HTTP Method

PATCH

Authentication

`Authorization: Bearer token`

Parameters

Parameter	Description
`allOrNone`	Optional. Indicates whether to roll back the entire request when the creation of any object fails (`true`) or to continue with the independent creation of other objects in the request. The default is `false`. Note: If the sObject Collections request is embedded in a Composite request, the Composite request's `allOrNone` parameter can also affect the results. See allOrNone Parameters in Composite and Collections Requests.
`records`	Required. A list of sObjects. In a PATCH request using sObject Collections, set the `type` attribute for each object. Don't set the `id` field for any object. Instead, use the external ID field specified in the request URI.

Example

Example Request

```
curl -X PATCH
https://MyDomainName.my.salesforce.com/services/data/v60.0/composite/sobjects/Account/MyExtId__c
 -H "Authorization: Bearer token" -H "Content-Type: application/json" -d
"@exampleRequestBody.json"
```

Example Request Body

```
{
    "allOrNone" : false,
    "records" : [{
        "attributes" : {"type" : "Account"},
        "Name" : "Company One",
        "MyExtId__c" : "AAA"
    }, {
        "attributes" : {"type" : "Account"},
        "Name" : "Company Two",
        "MyExtId__c" : "BBB"
    }]
}
```

Example Response Body

```
HTTP/1.1 200 OK
[
    {
        "id": "001xx0000004GxDAAU",
        "success": true,
        "errors": [],
        "created": true
    },
    {
        "id": "001xx0000004GxEAAU",
        "success": true,
        "errors": [],
        "created": false
    }
]
```

Example Response Body (Some Items Failed and `allOrNone` is `false`)

```
HTTP/1.1 200 OK

[
   {
      "id" : "001xx0000004GxDAAU",
      "success" : true,
      "errors" : [ ]
   },
   {
      "success" : false,
      "errors" : [
         {
```

```
                "statusCode" : "MALFORMED_ID",
                "message" : "Contact ID: id value of incorrect type: 001xx0000004GxEAAU",
                "fields" : [
                   "Id"
                ]
             }
          ]
       }
]
```

Example Response Body (Some Items Failed and `allOrNone` is `true`)

```
HTTP/1.1 200 OK

[
   {
      "id" : "001xx0000004GxDAAU",
      "success" : false,
      "errors" : [
         {
            "statusCode" : "ALL_OR_NONE_OPERATION_ROLLED_BACK",
            "message" : "Record rolled back because not all records were valid and the
 request was using AllOrNone header",
            "fields" : [ ]
         }
      ]
   },
   {
      "success" : false,
      "errors" : [
         {
            "statusCode" : "MALFORMED_ID",
            "message" : "Contact ID: id value of incorrect type: 001xx0000004GxEAAU",
            "fields" : [
               "Id"
            ]
         }
      ]
   }
]
```

SEE ALSO:

sObject Rows by External ID

Delete Records Using sObject Collections

Use a DELETE request with sObject Collections to delete up to 200 records, returning a list of DeleteResult objects. You can choose to roll back the entire request when an error occurs.

- The DeleteResult objects are returned in the order in which the IDs of the deleted objects were specified.
- You can't delete records for multiple object types in one call when one of those types is related to a feature in the Salesforce Setup area.

If the request isn't well formed, the API returns a `400 Bad Request` HTTP Status. Fix the syntax of the request and try again. If the request is well formed, the API returns a `200 OK` HTTP Status. If an item was processed successfully, the `success` flag shows for that item. Error information is returned in the `errors` array.

Syntax

URI

`/services/data/vXX.X/composite/sobjects?ids=recordId,recordId`

Formats

JSON, XML

HTTP Method

DELETE

Authentication

`Authorization: Bearer token`

Parameters

Parameter	Description
`allOrNone`	Optional. Indicates whether to roll back the entire request when the deletion of any object fails (`true`) or to continue with the independent deletion of other objects in the request. The default is `false`. Note: If the sObject Collections request is embedded in a Composite request, the Composite request's `allOrNone` parameter can also affect the results. See allOrNone Parameters in Composite and Collections Requests.
`ids`	Required. A list of up to 200 IDs of objects to be deleted. The IDs can belong to different object types, including custom objects.

Example

Example Request

```
curl -X DELETE
https://MyDomainName.my.salesforce.com/services/data/v60.0/composite/sobjects?ids=001xx000003DGb2AAG,003xx000004TmiQAAS&allOrNone=false
 -H "Authorization: Bearer token"
```

Example Response Body

```
HTTP/1.1 200 OK

[
   {
      "id" : "001RM000003oLrHYAU",
      "success" : true,
      "errors" : [ ]
   },
   {
      "id" : "001RM000003oLraYAE",
      "success" : true,
```

```
      "errors" : [ ]
   }
]
```

Example Response Body (Some Items Failed and `allOrNone` is `false`)

```
HTTP/1.1 200 OK

[
   {
      "id" : "001RM000003oLrfYAE",
      "success" : true,
      "errors" : [ ]
   },
   {
      "success" : false,
      "errors" : [
         {
            "statusCode" : "MALFORMED_ID",
            "message" : "malformed id 001RM000003oLrB000",
            "fields" : [ ]
         }
      ]
   }
]
```

Example Response Body (Some Items Failed and `allOrNone` is `true`)

```
HTTP/1.1 200 OK

[
   {
      "id" : "001RM000003oLruYAE",
      "success" : false,
      "errors" : [
         {
            "statusCode" : "ALL_OR_NONE_OPERATION_ROLLED_BACK",
            "message" : "Record rolled back because not all records were valid and the
 request was using AllOrNone header",
            "fields" : [ ]
         }
      ]
   },
   {
      "success" : false,
      "errors" : [
         {
            "statusCode" : "MALFORMED_ID",
            "message" : "malformed id 001RM000003oLrB000",
            "fields" : [ ]
         }
      ]
   }
]
```