
Force.com Tooling API

Developer's Guide

Version 34.0, Summer '15



CONTENTS

Chapter 1: Introducing Tooling API	1
When to Use Tooling API	1
Use Tooling API with REST	2
Use Tooling API with SOAP	5
Objects, Namespaces, and Tooling API	11
System Fields	12
Programming Objects	13
Setup Objects	14
Tooling Objects	16
Operational Objects	17
Chapter 2: Tooling API Objects	18
ApexClass	22
ApexClassMember	23
ApexCodeCoverage	26
ApexCodeCoverageAggregate	28
ApexComponent	29
ApexComponentMember	30
ApexExecutionOverlayAction	32
ApexExecutionOverlayResult	34
ApexLog	37
ApexOrgWideCoverage	39
ApexPage	40
ApexPageMember	40
ApexResult	43
ApexTestQueueItem	43
ApexTestResult	48
ApexTrigger	50
ApexTriggerMember	54
AuraDefinition	56
AuraDefinitionBundle	58
BusinessProcess	60
CompactLayout	61
CompactLayoutItemInfo	62
CompactLayoutInfo	64
ContainerAsyncRequest	68
CustomField	70
CustomFieldMember	72
CustomObject	74

Contents

CustomTab	77
DataType	80
DeployDetails	82
EmailTemplate	83
EntityDefinition	85
EntityLimit	96
EntityParticle	98
FieldDefinition	105
FieldSet	124
FlexiPage	125
Flow	127
FlowDefinition	130
HeapDump	133
HistoryRetentionJob	134
Layout	136
LookupFilter	138
MenuItem	141
MetadataContainer	144
Profile	145
ProfileLayout	146
Publisher	147
QueryResult	150
QuickActionDefinition	151
QuickActionList	156
QuickActionListItem	157
RecentlyViewed	158
RecordType	163
RelationshipDomain	165
RelationshipInfo	168
SearchLayout	170
ServiceFieldDataType	174
Scontrol	174
SOQLResult	176
StandardAction	177
StaticResource	181
SymbolTable	182
TraceFlag	186
User	191
UserEntityAccess	193
ValidationRule	195
WebLink	199
WorkflowAlert	205
WorkflowFieldUpdate	207
WorkflowOutboundMessage	209

Contents

WorkflowRule 211

WorkflowTask 212

Chapter 3: AllOrNoneHeader 215

Index 218

CHAPTER 1 Introducing Tooling API

Tooling API provides SOAP and REST interfaces that allow you to build custom development tools for Force.com applications.

For example, you can:

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

Tooling API exposes metadata used in developer tooling that you can access through REST or SOAP. Use Tooling API like the Salesforce REST API and SOAP API with one of the Tooling WSDLs.

For detailed descriptions of Tooling API objects and the REST resources and SOAP calls that each object supports, see [Tooling API Objects](#).

IN THIS SECTION:

[When to Use Tooling API](#)

[Use Tooling API with REST](#)

[Use Tooling API with SOAP](#)

[Objects, Namespaces, and Tooling API](#)

Tooling API objects provide programmatic access to data and metadata.

When to Use Tooling API

Tooling API facilitates the development of any organization-based user interface or user experience that requires create, read, update, or delete operations to Salesforce metadata. By contrast, Metadata API facilitates the migration of Salesforce metadata from one organization to another.

Perform CRUD Operations on Salesforce Metadata

To perform the following tasks, use Tooling API:

- Manage working copies of Apex classes and triggers and Visualforce pages and components using the [ApexClassMember](#), [ApexTriggerMember](#), [ApexPageMember](#), [ApexComponentMember](#), and [MetadataContainer](#) objects.
- Manage working copies of static resource files using the [StaticResource](#) object.
- Check for updates and errors in working copies of Apex classes and triggers and Visualforce pages and components using the [ContainerAsyncRequest](#) object.
- Commit changes to your organization using the [ContainerAsyncRequest](#) object.
- Set heap dump markers using the [ApexExecutionOverlayAction](#) object.
- Overlay Apex code or SOQL statements on an Apex execution using the [ApexExecutionOverlayAction](#) object.
- Execute anonymous Apex. For sample code, see [Use Tooling API with SOAP](#) and [Use Tooling API with REST](#).
- To generate log files for yourself or for other users, set checkpoints with the [TraceFlag](#) object.
- Access debug log and heap dump files using the [ApexLog](#) and [ApexExecutionOverlayResult](#) objects.

- Manage custom fields on custom objects using the [CustomField](#) object.
- Access code coverage results using the [ApexCodeCoverage](#), [ApexOrgWideCoverage](#), and [ApexCodeCoverageAggregate](#) objects.
- Execute tests, and manage test results using the [ApexTestQueueItem](#) and [ApexTestResult](#) objects.
- Manage validation rules and workflow rules using the [ValidationRule](#) and [WorkflowRule](#) objects.

Select the Right Protocol for Your Application

Tooling API provides supports both SOAP and REST.

- Use REST API if you're using a language that isn't strongly typed, like JavaScript. See [Use Tooling API with REST](#).
- Use SOAP API if you're using a strongly typed language like Java that generates Web service client code. See [Use Tooling API with SOAP](#).

Use Tooling API with REST

Use REST API if you're using a language that isn't strongly typed, like JavaScript. The Tooling REST API can be used just like the Force.com REST API; for details on usage, syntax, and authentication, see the [Force.com REST API Developer's Guide](#).

Resources

This section lists supported REST resources in Tooling API.

The base URI for each Tooling REST API resource is `http://domain/services/data/vXX.X/tooling/` where *domain* is a Salesforce instance or a custom domain and *vXX.X* is the API version number. For example:

`http://na1.salesforce.com/services/data/v28.0/tooling/`

Like the Force.com REST API, Tooling API uses the following resources.

URI	Supported Methods	Description
<code>/completions?type=</code>	GET	Retrieves available code completions of the referenced type. Currently only supports Apex system method symbols (<code>type=apex</code>). Available from API version 28.0 or later.
<code>/executeAnonymous/?anonymousBody=<url encoded body></code>	GET	Executes Apex code anonymously. Available from API version 29.0 or later.
<code>/query/?q=</code>	GET	<p>Executes a query against a Tooling API object and returns data that matches the specified criteria.</p> <p>If the query results are too large, the response contains the first batch of results and a query identifier. The identifier can be used in a request to retrieve the next batch.</p>

URI	Supported Methods	Description
/runTestsAsynchronous/?classids=<comma separated list of class IDs>	GET	Executes the tests in the specified classes. Running tests asynchronously allows methods to process in parallel, cutting down your test run times.
<pre>/runTestsAsynchronous/ Body: {"classids":"<comma-separated list of class IDs>"}</pre> <p>or</p> <pre>/runTestsAsynchronous/ Body: {"tests":<tests array>}</pre>	POST	<p>Runs one or more methods within one or more Apex classes, using the asynchronous test execution mechanism.</p> <p><tests array> is an array of objects, each with two parameters: <code>classId</code> and <code>testMethods</code>. The <code>classId</code> parameter is a String holding a test class's ID. The <code>testMethods</code> parameter is an array of Strings, each containing the name of a test method in the test class. Each class ID can be listed only once. Multiple occurrences of a test method name in a <code>testMethods</code> array are ignored. Test methods that don't exist are skipped. A null or missing <code>testMethods</code> array specifies that all test methods in the test class are run.</p> <p>Example <tests array>:</p> <pre>[{ "classId" : "<classId 1>", "testMethods" : ["testMethod1", "testMethod2", "testMethod3"] }, { "classId" : "<classId 2>", "testMethods" : ["testMethod1", "testMethod2"] }];</pre>
/runTestsSynchronous/?classnames=<comma-separated list of class names>	GET	Executes the tests in the specified classes using the synchronous test execution mechanism.
/subjects/	GET	Lists the available Tooling API objects and their metadata.
/subjects/ SubjectName /	GET POST	<p>Describes the individual metadata for the specified object or creates a record for a given object.</p> <ul style="list-style-type: none"> To retrieve the metadata for the <code>ApexExecutionOverlayAction</code> object, use the GET method. To create a <code>ApexExecutionOverlayAction</code> object, use the POST method.
/subjects/ SubjectName /describe/	GET	<p>Completely describes the individual metadata at all levels for the specified object.</p> <p>For example, use this resource to retrieve the fields, URLs, and child relationships for a Tooling API object.</p>

URI	Supported Methods	Description
/subjects/ ObjectName / id /	GET PATCH DELETE	Accesses records based on the specified object ID. Use the GET method to retrieve records or fields, the DELETE method to delete records, and the PATCH method to update records.
/subjects/ ApexLog / id /Body/	GET	Retrieves a raw debug log by ID. Available from API version 28.0 or later.

Starting in API version 34.0, the following resources support the [AllOrNoneHeader](#) header.

- /subjects/**ObjectName**/ (POST method)
- /subjects/**ObjectName**/**id**/ (PATCH and DELETE methods)

By default, if `AllOrNoneHeader` isn't used, REST calls can save a partial set of records for records with no errors (equivalent to `AllOrNoneHeader=false`). If `AllOrNoneHeader` is set to `true`, records are saved only when there are no failures in any record in the request.

Examples

The following examples use Apex to execute REST requests, but you can use any standard REST tool to access Tooling REST API.



Note: Salesforce runs on multiple server instances. The examples in this guide use the *na1* instance. Be sure to use your organization's instance name.

First, set up the connection to your org and the HTTP request type:

```
HttpRequest req = new HttpRequest();
req.setHeader('Authorization', 'Bearer ' + UserInfo.getSessionID());
req.setHeader('Content-Type', 'application/json');
```

At the end of each request, add the following code to send the request and retrieve the body of the response:

```
Http h = new Http();
HttpResponse res = h.send(req);
system.debug(res.getBody());
```

To get a description of all available objects in Tooling API:

```
req.setEndpoint('http://na1.salesforce.com/services/data/v28.0/tooling/subjects/');
req.setMethod('GET');
```

To get a description of a specific Tooling API object, for example [TraceFlag](#):

```
req.setEndpoint('http://na1.salesforce.com/services/data/v28.0/tooling/subjects/TraceFlag/');
req.setMethod('GET');
```

To get a description of all the metadata for a specific Tooling API object, for example [TraceFlag](#):

```
req.setEndpoint('http://na1.salesforce.com/services/data/v28.0/tooling/subjects/TraceFlag/describe/');
req.setMethod('GET');
```

To create a new Tooling API object, for example [MetadataContainer](#):

```
req.setEndpoint('http://na1.salesforce.com/services/data/v28.0/tooling/subjects/
MetadataContainer/');
req.setBody('{"Name":"TestContainer"}');
req.setMethod('POST');
```



Tip: Use the ID from this call in the rest of the examples.

To retrieve a Tooling API object by ID, for example [MetadataContainer](#):

```
req.setEndpoint('http://na1.salesforce.com/services/data/v28.0/tooling/subjects/
MetadataContainer/ + containerID + '/');
req.setMethod('GET');
```

To update a Tooling API object by ID, for example [MetadataContainer](#):

```
req.setEndpoint('http://na1.salesforce.com/services/data/v28.0/tooling/subjects/
MetadataContainer/ + containerID + '/');
req.setBody('{"Name":"NewlyNamedContainer"}');
req.setMethod('PATCH');
```

To query a Tooling API object by ID, for example [MetadataContainer](#):

```
req.setEndpoint('http://na1.salesforce.com/services/data/v28.0/tooling/query/?q=
Select+id,Name+from+MetadataContainer+Where+ID=\' + containerID + \'');
req.setMethod('GET');
```

To query an object within a [MetadataContainer](#):

```
req.setEndpoint('http://na1.salesforce.com/services/data/v28.0/tooling/query/?q=
Select+id,Body,LastSyncDate,Metadata+from+ApexClassMember+Where+MetadataContainerID=\'
+ containerID + \'');
req.setMethod('GET');
```

To check on the status of a deployment, using [ContainerAsyncRequest](#):

```
req.setEndpoint('http://na1.salesforce.com/services/data/v28.0/tooling/subjects/
ContainerAsyncRequest/' + requestID + '/');
req.setMethod('GET');
```

To execute anonymous Apex:

```
req.setEndpoint('http://na1.salesforce.com/services/data/v28.0/tooling/executeAnonymous/?
anonymousBody=System.debug('Test')%3B');
req.setMethod('GET');
```

To retrieve your Apex classes and triggers, and the global Apex classes and triggers from your installed managed packages:

```
req.setEndpoint('http://na1.salesforce.com/services/data/v33.0/tooling/apexManifest');
req.setMethod('GET');
```

Use Tooling API with SOAP

Use SOAP API if you're using a strongly typed language like Java that generates Web service client code. Tooling SOAP API is used just like the Salesforce SOAP API. For details on usage, syntax, and authentication, see the [SOAP API Developer's Guide](#).

To access the Tooling API WSDL, from Setup, click **Develop > API** and click **Generate Tooling WSDL**.

Like the Salesforce SOAP API, Tooling API uses the following calls.

Call	Description
<code>create()</code>	Adds one or more records to your organization's data.
<code>delete()</code>	Deletes one or more records from your organization's data.
<code>describeGlobal()</code>	Lists the available Tooling API objects and their metadata.
<code>describeSObjects()</code>	Describes metadata (field list and object properties) for the specified object or array of objects. Call <code>describeGlobal()</code> to retrieve a list of all Tooling API objects for your organization, then iterate through the list and use <code>describeSObjects()</code> to obtain metadata about individual objects.
<code>executeAnonymous(string apexcode)</code>	Executes the specified block of Apex anonymously and returns the result.
<code>query()</code>	Executes a query against a Tooling API object and returns data that matches the specified criteria.
<code>queryMore()</code>	Retrieves the next batch of objects from a <code>query()</code> .
<code>retrieve()</code>	Retrieves one or more records based on the specified IDs.
<code>runTests()</code> and <code>runTestsAsynchronous()</code>	Executes test methods in the specified classes. Running tests asynchronously allows methods to process in parallel, cutting down your test run times. For example code, see ApexTestQueueItem .
<code>update()</code>	Updates one or more existing records in your organization's data.
<code>upsert()</code>	Creates records and updates existing records; uses a custom field to determine the presence of existing records.

Starting in API version 34.0, the following calls support the [AllOrNoneHeader](#) header: `create()`, `update()`, `upsert()`, and `delete()`.

By default, if `AllOrNoneHeader` isn't used, those calls can save a partial set of records for records with no errors (equivalent to `AllOrNoneHeader=false`). If `AllOrNoneHeader` is set to `true`, records are saved only when there are no failures in any record in the call.

Examples

These examples use Java, but you can use Tooling SOAP API in any language that supports Web services.

To compile Apex classes or triggers in Developer Edition or sandbox organizations, use `create()`. The next sample uses [ApexClass](#) to compile a simple class with a single method called `SayHello`.

```
String classBody = "public class Messages {\n"
    + "public string SayHello() {\n"
    + "    return 'Hello';\n" + "}\n"
    + "}";

// create an ApexClass object and set the body
ApexClass apexClass = new ApexClass();
```

```

apexClass.Body = classBody;
ApexClass[] classes = { apexClass };

// call create() to add the class
SaveResult[] saveResults = sforce.create(classes);
for (int i = 0; i < saveResults.Length; i++)
{
    if (saveResults[i].success)
    {
        Console.WriteLine("Successfully created Class: " +
            saveResults[i].id);
    }
    else
    {
        Console.WriteLine("Error: could not create Class ");
        Console.WriteLine("    The error reported was: " +
            saveResults[i].errors[0].message + "\n");
    }
}

```

The `IsCheckOnly` parameter on [ContainerAsyncRequest](#) indicates whether an asynchronous request compiles code but doesn't execute or save it (`true`), or compiles and save the code (`false`).

The next example expands upon the first by modifying the `SayHello()` method to accept a person's first and last name. This example uses [MetadataContainer](#) with [ApexClassMember](#) to retrieve and update the class, and [ContainerAsyncRequest](#) to compile and deploy the changes to the server. You can use the same method with [ApexTriggerMember](#), [ApexComponentMember](#), and [ApexPageMember](#).



Note: To test your code, modify the `IsCheckOnly` parameter in the next sample, and log in to your organization after a successful execution to verify the results.

- When `IsCheckOnly = true`, the `SayHello()` method remains the same. `ApexClassMember` contains the compiled results, but the class on the server remains the same.
- When `IsCheckOnly = false`, the `SayHello()` method shows the change to accept a person's first and last name.

```

String updatedClassBody = "public class Messages {\n"
    + "public string SayHello(string fName, string lName) {\n"
    + "    return 'Hello ' + fName + ' ' + lName;\n"
    + "}" + "\n";

//create the metadata container object
MetadataContainer Container = new MetadataContainer();
Container.Name = "SampleContainer";

MetadataContainer[] Containers = { Container };
SaveResult[] containerResults = sforce.create(Containers);
if (containerResults[0].success)
{
    String containerId = containerResults[0].id;

    //create the ApexClassMember object
    ApexClassMember classMember = new ApexClassMember();
    //pass in the class ID from the first example
    classMember.ContentEntityId = classId;
    classMember.Body = updatedClassBody;
    //pass the ID of the container created in the first step

```

```

classMember.MetadataContainerId = containerId;
ApexClassMember[] classMembers = { classMember };

SaveResult[] MembersResults = sforce.create(classMembers);
if (MembersResults[0].success)
{
    //create the ContainerAsyncRequest object
    ContainerAsyncRequest request = new ContainerAsyncRequest();
    //if the code compiled successfully, save the updated class to the server
    //change to IsCheckOnly = true to compile without saving
    request.IsCheckOnly = false;
    request.MetadataContainerId = containerId;
    ContainerAsyncRequest[] requests = { request };
    SaveResult[] RequestResults = sforce.create(requests);
    if (RequestResults[0].success)
    {
        string requestId = RequestResults[0].id;

        //poll the server until the process completes
        QueryResult queryResult = null;
        String soql = "SELECT Id, State, ErrorMsg FROM ContainerAsyncRequest where id
= '" + requestId + "'";
        queryResult = sforce.query(soql);
        if (queryResult.size > 0)
        {
            ContainerAsyncRequest _request =
(ContainerAsyncRequest)queryResult.records[0];
            while (_request.State.ToLower() == "queued")
            {
                //pause the process for 2 seconds
                Thread.Sleep(2000);

                //poll the server again for completion
                queryResult = sforce.query(soql);
                _request = (ContainerAsyncRequest)queryResult.records[0];
            }

            //now process the result
            switch (_request.State)
            {
                case "Invalidated":
                    break;

                case "Completed":
                    //class compiled successfully
                    //see the next example on how to process the SymbolTable
                    break;

                case "Failed":
                    . . . break;

                case "Error":
                    break;
            }
        }
    }
}

```

```

        case "Aborted":
            break;

        }
    }
    else
    {
        //no rows returned
    }
}
else
{
    Console.WriteLine("Error: could not create ContainerAsyncRequest object");

    Console.WriteLine("    The error reported was: " +
        RequestResults[0].errors[0].message + "\n");
}
}
else
{
    Console.WriteLine("Error: could not create Class Member ");
    Console.WriteLine("    The error reported was: " +
        MembersResults[0].errors[0].message + "\n");
}
}
else
{
    .. Console.WriteLine("Error: could not create MetadataContainer ");
    Console.WriteLine("    The error reported was: " +
        containerResults[0].errors[0].message + "\n");
}
}
}

```

To access Apex class and trigger data in a structured format, use a [SymbolTable](#).

The next sample queries the [ApexClassMember](#) object created in the previous example to obtain the [SymbolTable](#) of the modified class.

 **Note:** The SOQL statement used depends on when the data is retrieved.

- To execute the query from within the previous sample, use the ID of the [ContainerAsyncRequest](#). For example, `SELECT Body, ContentEntityId, SymbolTable FROM ApexClassMember where MetadataContainerId = ' " + requestId + "'`
- Otherwise, use the ID of the modified class as shown in the next sample. For example, `SELECT ContentEntityId, SymbolTable FROM ApexClassMember where ContentEntityId = ' " + classId + "'`

```

//use the ID of the class from the previous step
string classId = "01pA00000036itIIAQ";
QueryResult queryResult = null;
String soql = "SELECT ContentEntityId, SymbolTable FROM ApexClassMember where
ContentEntityId = ' " + classId + "'";

queryResult = sforce.query(soql);
if (queryResult.size > 0)
{
    ApexClassMember apexClass = (ApexClassMember)queryResult.records[0];
}

```

```

SymbolTable symbolTable = apexClass.SymbolTable;

foreach (Method _method in symbolTable.methods)
{
    //here's the SayHello method
    String _methodName = _method.name;

    //is the method Global, Public or Private?
    String _methodVisibility = _method.visibility.ToString();

    //get the method's return type
    string _methodReturnType = _method.returnType;

    //get the fName & lName parameters
    foreach (Parameter _parameter in _method.parameters)
    {
        string _paramName = _parameter.name;
        string _parmType = _parameter.type;
    }
}
else
{
    //unable to locate class
}

```

To add checkpoints to your code for debugging, use [ApexExecutionOverlayAction](#).

This sample adds a checkpoint to the class from the previous samples:

```

//use the ID of the class from the first sample.
string classId = "01pA00000036itIIAQ";

ApexExecutionOverlayAction action = new ApexExecutionOverlayAction();
action.ExecutableEntityId = classId;
action.Line = 3;
action.LineSpecified = true;
action.Iteration = 1;
action.IterationSpecified = true;
ApexExecutionOverlayAction[] actions = { action };

SaveResult[] actionResults = sforce.create(actions);
if (actionResults[0].success)
{
    // checkpoint created successfully
}
else
{
    Console.WriteLine("Error: could not create Checkpoint ");
    Console.WriteLine("    The error reported was: " +
        actionResults[0].errors[0].message + "\n");
}

```


Objects, Namespaces, and Tooling API

Tooling API objects provide programmatic access to data and metadata.

The Tooling API WSDL includes the tooling namespace `tns (urn:tooling.soap.sforce.com)` and the metadata namespace `mns (urn:metadata.tooling.soap.sforce.com)`. Some objects or types in the `mns` namespace occur in both the Metadata API WSDL and the Tooling API WSDL.

- If objects or types are identical in the Tooling API WSDL and Metadata API WSDL, they are documented in the Metadata API Developer's Guide.
- If objects or types are different in the Tooling API WSDL, or only occur in the Tooling API WSDL, they are documented here.

To verify the complete list of fields for an object, access the Tooling API WSDL:

1. Navigate to Setup.
2. Select **Develop > API**.
3. Select either **Generate Tooling WSDL** or **Generate Tooling WSDL With Strongly Typed Enums**.



Note: Frequently occurring system fields are described in [System Fields](#) on page 12.

SOQL Operation Limitations in Tooling API

The following objects in Tooling API don't support SOQL operations `COUNT()`, `GROUP BY`, `LIMIT`, `LIMIT OFFSET`, `OR`, and `NOT`.

- CompactLayoutInfo
- CompactLayoutItemInfo
- DataType
- EntityDefinition
- EntityLimit
- EntityParticle
- FieldDefinition
- PicklistValueInfo
- Publisher
- RelationshipDomain
- RelationshipInfo
- SearchLayout
- ServiceFieldDataType
- StandardAction
- UserEntityAccess
- UserFieldAccess

The unsupported operations for these objects return errors or incorrect results as these examples illustrate.

COUNT()

Example Query: `SELECT COUNT() FROM EntityDefinition`

Error Returned: The "DurableId" field is of type number, but the value from the external data source is "Account.SearchResult". Contact your administrator for help.

GROUP BY

Example Query: `SELECT COUNT(qualifiedapiname), isfeedenabled FROM EntityDefinition GROUP BY isfeedenabled`

Error Returned: The requested operation is not yet supported by this SObject storage type, contact salesforce.com support for more information.

LIMIT, LIMIT OFFSET

Example Queries:

`SELECT qualifiedapiname FROM EntityDefinition LIMIT 5`

`SELECT qualifiedapiname FROM EntityDefinition LIMIT 5 OFFSET 10`

An incorrect result is returned because LIMIT and LIMIT OFFSET are ignored.

NOT

Example Query: `SELECT qualifiedapiname FROM EntityDefinition WHERE qualifiedapiname!='Account'`

Error Returned: Only equals comparisons permitted

OR

Example Query: `SELECT qualifiedapiname, keyprefix FROM EntityDefinition WHERE isdeletable=true OR (isfeedenabled=false AND keyprefix='01j')`

Error Returned: Disjunctions not supported

Quick-Reference Tables

Look up information about system fields that occur on most objects and look up objects by the type of work they do.

IN THIS SECTION:

[System Fields](#)

Some fields are system-generated. They are on most Tooling API objects, and are read-only.

[Programming Objects](#)

Use programming objects to interact with programmatic artifacts: Apex, Visualforce, and Lightning.

[Setup Objects](#)

Use setup objects to interact with metadata for declarative development. For example, you may wish to create your own version of Setup, or restrict the amount of data you need to push to an app on a mobile phone.

[Tooling Objects](#)

Use these objects to build tools around test results, debugging, code coverage, and more.

[Operational Objects](#)

Use the following objects for Tooling API operations.

System Fields

Some fields are system-generated. They are on most Tooling API objects, and are read-only.

These fields are automatically updated during API operations. For example, the `Id` field is automatically generated when a record is created, and the `LastModifiedDate` is automatically updated during any operation on an object.

Field	Field Type	Description
<code>Id</code>	ID	Globally unique string that identifies a record. <code>Id</code> fields have Defaulted on create and Filter access.
<code>IsDeleted</code>	boolean	Indicates whether the record has been moved to the Recycle Bin (<code>true</code>) or not (<code>false</code>). Because this field does not appear in all objects, it is listed in the field table for each object.
<code>CreatedBy</code>	User	The user who created the record. <code>CreatedBy</code> fields have Defaulted on create, Filter, Group, and Sort access.
<code>CreatedById</code>	reference	ID of the User who created this record. <code>CreatedById</code> fields have Defaulted on create, Filter, Group, and Sort access.
<code>CreatedDate</code>	dateTime	Date and time when this record was created. <code>CreatedDate</code> fields have Defaulted on create, Filter, and Sort access.
<code>LastModifiedBy</code>	User	The user who last modified this record. <code>LastModifiedBy</code> fields have Defaulted on create, Filter, Group, and Sort access.
<code>LastModifiedById</code>	reference	ID of the User who last updated this record. <code>LastModifiedById</code> fields have Defaulted on create, Filter, Group, and Sort access.
<code>LastModifiedDate</code>	dateTime	Date and time when a user last modified this record. <code>LastModifiedDate</code> fields have Defaulted on create, Filter, and Sort access.
<code>SystemModstamp</code>	dateTime	Date and time when this record was last modified by a user or by an automated process (such as a trigger). <code>SystemModstamp</code> fields have Defaulted on create and Filter access.

To verify which fields are available for an object, check the Tooling API WSDL.

Programming Objects

Use programming objects to interact with programmatic artifacts: Apex, Visualforce, and Lightning.

ApexClass

Represents the saved copy of an Apex class. `ApexClass` uses the cached version of the class unless one is unavailable.

ApexClassMember

Represents the working copy of an Apex class for editing, saving or compiling in a `MetadataContainer`.

ApexComponent

Represents the saved copy of a Visualforce component. `ApexComponent` uses the cached version of the class unless one is unavailable.

ApexComponentMember

Represents the working copy of a Visualforce component for editing, saving or compiling in a `MetadataContainer`.

ApexPage

Represents the saved copy of an Apex page. `ApexPage` uses the cached version of the class unless one is unavailable.

ApexPageMember

Represents the working copy of a Visualforce page for editing, saving or compiling in a `MetadataContainer`.

ApexTrigger

Represents the saved copy of an Apex trigger. ApexTrigger uses the cached version of the class unless one is unavailable.

ApexTriggerMember

Represents the working copy of an Apex trigger for editing, saving or compiling in a MetadataContainer.

AuraDefinition

Represents a Lightning definition, such as component markup, a client-side controller, or an event. Lightning components is a beta feature.

AuraDefinitionBundle

Represents a Lightning definition bundle, such as a component or application bundle. A bundle contains a Lightning definition and all its related resources. Lightning components is a beta feature.

StaticResource

Represents the working copy of a static resource file for editing or saving. Static resources allow you to upload content that you can reference in a Visualforce page, including images, stylesheets, JavaScript, and other files.

Setup Objects

Use setup objects to interact with metadata for declarative development. For example, you may wish to create your own version of Setup, or restrict the amount of data you need to push to an app on a mobile phone.

BusinessProcess

Represents a business process.

CompactLayout

Represents the values that define a compact page layout.

CompactLayoutItemInfo

Represents a field selected for a compact layout, and the order of that field in the compact layout.

CompactLayoutInfo

Represents the metadata for a custom or standard compact layout.

CustomField

Represents a custom field on a custom object that stores data unique to your organization.

CustomFieldMember

Represents the working copy of a field for editing or saving in a MetadataContainer.

CustomObject

Represents a custom object that stores data unique to your organization. Includes access to the associated CustomObject object and related fields in Salesforce Metadata API.

CustomTab

Represents a custom tab.

EmailTemplate

Represents an email template.

EntityDefinition

Provides row-based access to metadata about standard and custom objects.

EntityLimit

Represents the limits for an object as displayed in the Setup UI.

FieldDefinition

Represents a standard or custom field, providing row-based access to field metadata. Contrast FieldDefinition with EntityParticle, which represents each element of a field that can be presented in a user interface.

FieldSet

Represents the metadata for a group of fields.

FlexiPage

Represents a Lightning Page. A Lightning Page is the home page for an app that appears as a menu item in the Salesforce1 navigation menu. Includes access to the associated FlexiPage object in the Salesforce Metadata API.

Flow

Use the Flow object to retrieve and update specific flow versions.

FlowDefinition

The parent of a set of flow versions.

HistoryRetentionJob

Represents the body of retained data from the archive, and the status of the archived data.

Layout

Represents a page layout. For more information, see “Manage Page Layouts” in the Salesforce online help.

LookupFilter

Represents a lookup filter, which restricts the valid values and lookup dialog results for lookup, master-detail, and hierarchical relationship fields.

MenuItem

Represents a menu item.

Profile

Represents a user profile. A profile defines a user's permission to perform different functions within Salesforce.

ProfileLayout

Represents a profile layout.

QuickActionDefinition

Represents the definition of a quick action.

QuickActionList

Represents a list of quick actions.

QuickActionListItem

Represents an item in a quick action list.

RecentlyViewed

Represents metadata entities typically found in Setup such as page layout definitions, workflow rule definitions, and email templates that the current user has recently viewed.

RecordType

Represents a custom record type.

SearchLayout

Represents a search layout defined for an object.

Scontrol

Represents a custom s-control, which is custom content that our system hosts, but client applications execute. An s-control can contain any type of content that you can display or run in a Web browser.

User

Represents a user. You can retrieve standard fields on User with the Tooling API, but custom fields can't be retrieved.

WebLink

Represents a custom link to a URL or S-control.

ValidationRule

Represents a validation rule or workflow rule which specifies the formula for when a condition is met.

WorkflowAlert

Represents a workflow alert. A workflow alert is an email generated by a workflow rule or approval process and sent to designated recipients.

WorkflowFieldUpdate

Represents a workflow field update.

WorkflowOutboundMessage

Represents an outbound message. An outbound message is a workflow, approval, or milestone action that sends the information you specify to an endpoint you designate, such as an external service. Outbound messaging is configured in the Salesforce setup menu. Then you must configure the external endpoint. You can create a listener for the messages using the SOAP API.

WorkflowRule

Represents a workflow rule that is used to fire off a specific workflow action when the specified criteria is met. Includes access to the associated WorkflowRule object in Salesforce Metadata API.

WorkflowTask

Represents a workflow task that is used to fire off a specific workflow action when the specified criteria is met. Includes access to the associated WorkflowRule object in Salesforce Metadata API.

Tooling Objects

Use these objects to build tools around test results, debugging, code coverage, and more.

ApexCodeCoverage

Represents code coverage test results for an Apex class or trigger.

ApexCodeCoverageAggregate

Represents aggregate code coverage test results for an Apex class or trigger. Available in Tooling API version 29.0 and later.

ApexExecutionOverlayAction

Specifies an Apex code snippet or SOQL query to execute at a specific line of code in an Apex class or trigger and optionally generate a heap dump.

ApexExecutionOverlayResult

Represents the result from the Apex code snippet or SOQL query defined in the associated ApexExecutionOverlayAction, and the resulting heap dump if one was returned.

ApexLog

Represents a debug log.

ApexOrgWideCoverage

Represents code coverage test results for an entire organization.

ApexResult

A complex type that represents the result of Apex code executed as part of an ApexExecutionOverlayAction, returned in an ApexExecutionOverlayResult.

ApexTestQueueItem

Represents a single Apex class in the Apex job queue.

HeapDump

A complex type that represents a heap dump in an ApexExecutionOverlayResult object.

SOQLResult

A complex type that represents the result of a SOQL query in an ApexExecutionOverlayResult object.

SymbolTable

A complex type that represents all user-defined tokens in the `Body` of an ApexClass, ApexClassMember, or ApexTriggerMember and their associated line and column locations within the `Body`.

TraceFlag

Represents a trace flag that triggers an Apex debug log at the specified logging level.

Operational Objects

Use the following objects for Tooling API operations.

ContainerAsyncRequest

Allows you to compile and asynchronously deploy a MetadataContainer object to your organization.

DeployDetails

A complex type that contains detailed XML for any compile errors reported in the asynchronous request defined by a ContainerAsyncRequest object.

MetadataContainer

Manages working copies of ApexClassMember, ApexTriggerMember, ApexPageMember, and ApexComponentMember objects, including collections of objects to be deployed together.

The following Tooling API objects are used internally by the Developer Console.

- IDEPerspective
- IDEWorkspace
- User.WorkspaceId

CHAPTER 2 Tooling API Objects

Tooling API includes the following objects:

IN THIS SECTION:

[ApexClass](#)

Represents the saved copy of an Apex class. ApexClass uses the cached version of the class unless one is unavailable. Available from API version 28.0 or later.

[ApexClassMember](#)

Represents the working copy of an Apex class for editing, saving or compiling in a MetadataContainer.

[ApexCodeCoverage](#)

Represents code coverage test results for an Apex class or trigger. Available in Tooling API version 29.0 and later.

[ApexCodeCoverageAggregate](#)

Represents aggregate code coverage test results for an Apex class or trigger. Available in Tooling API version 29.0 and later.

[ApexComponent](#)

Represents the saved copy of a Visualforce component. ApexComponent uses the cached version of the class unless one is unavailable. Available from API version 28.0 or later.

[ApexComponentMember](#)

Represents the working copy of a Visualforce component for editing, saving or compiling in a MetadataContainer.

[ApexExecutionOverlayAction](#)

Specifies an Apex code snippet or SOQL query to execute at a specific line of code in an Apex class or trigger and optionally generate a heap dump.

[ApexExecutionOverlayResult](#)

Represents the result from the Apex code snippet or SOQL query defined in the associated ApexExecutionOverlayAction, and the resulting heap dump if one was returned. Available from API version 28.0 or later.

[ApexLog](#)

Represents a debug log.

[ApexOrgWideCoverage](#)

Represents code coverage test results for an entire organization. Available in Tooling API version 29.0 and later.

[ApexPage](#)

Represents the saved copy of an Apex page. ApexPage uses the cached version of the class unless one is unavailable. Available from API version 28.0 or later.

[ApexPageMember](#)

Represents the working copy of a Visualforce page for editing, saving or compiling in a MetadataContainer.

[ApexResult](#)

A complex type that represents the result of Apex code executed as part of an ApexExecutionOverlayAction, returned in an ApexExecutionOverlayResult. Available from API version 28.0 or later.

[ApexTestQueueItem](#)

Represents a single Apex class in the Apex job queue. Available from API version 30.0 or later.

Tooling API Objects

[ApexTestResult](#)

Represents the result of an Apex test method execution. Available from API version 30.0 or later.

[ApexTrigger](#)

Represents the saved copy of an Apex trigger. ApexTrigger uses the cached version of the class unless one is unavailable. Available from API version 28.0 or later.

[ApexTriggerMember](#)

Represents the working copy of an Apex trigger for editing, saving or compiling in a MetadataContainer.

[AuraDefinition](#)

Represents a Lightning definition, such as component markup, a client-side controller, or an event. Lightning components is a beta feature. Available in API version 32.0 and later.

[AuraDefinitionBundle](#)

Represents a Lightning definition bundle, such as a component or application bundle. A bundle contains a Lightning definition and all its related resources. Lightning components is a beta feature. Available in API version 32.0 and later.

[BusinessProcess](#)

Represents a business process.

[CompactLayout](#)

Represents the values that define a compact page layout.

[CompactLayoutItemInfo](#)

Represents a field selected for a compact layout, and the order of that field in the compact layout.

[CompactLayoutInfo](#)

Represents the metadata for a custom or standard compact layout.

[ContainerAsyncRequest](#)

Allows you to compile and asynchronously deploy a MetadataContainer object to your organization.

[CustomField](#)

Represents a custom field on a custom object that stores data unique to your organization. Includes access to the associated CustomField object and related fields in Salesforce Metadata API. Available from API version 28.0 or later.

[CustomFieldMember](#)

Represents the working copy of a field for editing or saving in a MetadataContainer. This object is available in API version 33.0 and later.

[CustomObject](#)

Represents a custom object that stores data unique to your organization. Includes access to the associated CustomObject object and related fields in Salesforce Metadata API. Available from API version 31.0 or later.

[CustomTab](#)

Represents a custom tab.

[DataType](#)

Represents the datatype of a field. Use this object with EntityDefinition, EntityParticle, or FieldDefinition to simplify queries. Available in Tooling API version 34.0 and later.

[DeployDetails](#)

A complex type that contains detailed XML for any compile errors reported in the asynchronous request defined by a ContainerAsyncRequest object. Replaces the JSON field `CompilerErrors` in Tooling API version 31.0 and later.

[EmailTemplate](#)

Represents an email template.

Tooling API Objects

[EntityDefinition](#)

Provides row-based access to metadata about standard and custom objects.

[EntityLimit](#)

Represents the limits for an object as displayed in the Setup UI.

[EntityParticle](#)

Represents each element of a field that can be presented in a user interface. Contrast EntityParticle with FieldDefinition, which represents each element of a field defined in the Metadata API. Available in Tooling API version 34.0 and later.

[FieldDefinition](#)

Represents a standard or custom field, providing row-based access to field metadata. Contrast FieldDefinition with EntityParticle, which represents each element of a field that can be presented in a user interface.

[FieldSet](#)

Represents the metadata for a group of fields. Available from API version 33.0 or later.

[FlexiPage](#)

Represents a Lightning Page. A Lightning Page is the home page for an app that appears as a menu item in the Salesforce1 navigation menu. Includes access to the associated FlexiPage object in the Salesforce Metadata API. Available from API version 31.0 or later.

[Flow](#)

Use the Flow object to retrieve and update specific flow versions.

[FlowDefinition](#)

The parent of a set of flow versions.

[HeapDump](#)

A complex type that represents a heap dump in an ApexExecutionOverlayResult object. Available from API version 28.0 or later.

[HistoryRetentionJob](#)

Represents the body of retained data from the archive, and the status of the archived data. Available in API version 29.0 or later.

[Layout](#)

Represents a page layout. For more information, see “Manage Page Layouts” in the Salesforce online help.

[LookupFilter](#)

Represents a lookup filter, which restricts the valid values and lookup dialog results for lookup, master-detail, and hierarchical relationship fields.

[MenuItem](#)

Represents a menu item.

[MetadataContainer](#)

Manages working copies of ApexClassMember, ApexTriggerMember, ApexPageMember, and ApexComponentMember objects, including collections of objects to be deployed together.

[Profile](#)

Represents a user profile. A profile defines a user's permission to perform different functions within Salesforce.

[ProfileLayout](#)

Represents a profile layout.

[Publisher](#)

Represents the publisher of objects and fields. For example, Salesforce is the publisher for standard objects, the organization is the publisher for custom objects, and the package is the publisher for installed packages. Available in Tooling API version 34.0 and later.

Tooling API Objects

[QueryResult](#)

Represents the results of a query. For example, if you query on the object `EntityDefinition`, all the layouts for that entity are returned as an array of `QueryResult` objects in the `Layouts` field. Available in Tooling API version 34.0 and later.

[QuickActionDefinition](#)

Represents the definition of a quick action.

[QuickActionList](#)

Represents a list of quick actions.

[QuickActionListItem](#)

Represents an item in a quick action list.

[RecentlyViewed](#)

Represents metadata entities typically found in Setup such as page layout definitions, workflow rule definitions, and email templates that the current user has recently viewed.

[RecordType](#)

Represents a custom record type.

[RelationshipDomain](#)

Represents the relationship an object has with other objects. Use this object in queries to answer the question “which objects are the child objects for the object defined in `ParentSubject`.” Available in Tooling API version 34.0 and later.

[RelationshipInfo](#)

Represents the properties of a relationship between objects. Use this object in queries to answer the question “which objects are parent objects for the object defined in `ChildSubject`.” Available in Tooling API version 34.0 and later.

[SearchLayout](#)

Represents a search layout defined for an object.

[ServiceFieldType](#)

Don't use this object.

[Scontrol](#)

Represents a custom s-control, which is custom content that our system hosts, but client applications execute. An s-control can contain any type of content that you can display or run in a Web browser.

[SOQLResult](#)

A complex type that represents the result of a SOQL query in an `ApexExecutionOverlayResult` object. Available from API version 28.0 or later.

[StandardAction](#)

Represents the buttons, links, and actions (standard actions) for a standard or custom object. This object is available in API version 34.0 and later.

[StaticResource](#)

Represents the working copy of a static resource file for editing or saving. Static resources allow you to upload content that you can reference in a Visualforce page, including images, stylesheets, JavaScript, and other files. Available in Tooling API version 29.0 and later.

[SymbolTable](#)

A complex type that represents all user-defined tokens in the `Body` of an `ApexClass`, `ApexClassMember`, or `ApexTriggerMember` and their associated line and column locations within the `Body`.

[TraceFlag](#)

Represents a trace flag that triggers an Apex debug log at the specified logging level.

[User](#)

Represents a user. You can retrieve standard fields on User with the Tooling API, but custom fields can't be retrieved.

[UserEntityAccess](#)

Represents the access that the current user has to an object. Available in Tooling API version 34.0 and later.

[ValidationRule](#)

Represents a validation rule or workflow rule which specifies the formula for when a condition is met. Available from API version 34.0 or later.

[WebLink](#)

Represents a custom link to a URL or S-control. Available in the Tooling API from API version 34.0 or later.

[WorkflowAlert](#)

Represents a workflow alert. A workflow alert is an email generated by a workflow rule or approval process and sent to designated recipients.

[WorkflowFieldUpdate](#)

Represents a workflow field update.

[WorkflowOutboundMessage](#)

Represents an outbound message. An outbound message is a workflow, approval, or milestone action that sends the information you specify to an endpoint you designate, such as an external service. Outbound messaging is configured in the Salesforce setup menu. Then you must configure the external endpoint. You can create a listener for the messages using the SOAP API.

[WorkflowRule](#)

Represents a workflow rule that is used to fire off a specific workflow action when the specified criteria is met. Includes access to the associated WorkflowRule object in Salesforce Metadata API.

[WorkflowTask](#)

Represents a workflow task that is used to fire off a specific workflow action when the specified criteria is met. Includes access to the associated WorkflowRule object in Salesforce Metadata API.

ApexClass

Represents the saved copy of an Apex class. ApexClass uses the cached version of the class unless one is unavailable. Available from API version 28.0 or later.

To edit, save, or compile Apex classes, use [ApexClassMember](#).

Supported SOAP API Calls

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

Supported REST API HTTP Methods

Query, GET, POST, PATCH, DELETE

Fields

Field Name	Details
SymbolTable	<p>Type SymbolTable</p> <p>Properties Nillable</p> <p>Description A complex type that represents all user-defined tokens in the <code>Body</code> of an ApexClass, ApexClassMember, or ApexTriggerMemeber and their associated line and column locations within the <code>Body</code>. This field is null if the symbol table cannot be created.</p>

Usage

To retrieve information about an Apex class, create an ApexClass object that references it. For example code, see [Use Tooling API with SOAP](#).

To edit, save, or compile Apex classes, use [ApexClassMember](#).



Note: If there is not a cached version of [SymbolTable](#), it will be compiled in the background and the query might take longer than expected. The SymbolTable returned from ApexClass does not contain references; to retrieve a SymbolTable with references, use [ApexClassMember](#).

ApexClassMember

Represents the working copy of an Apex class for editing, saving or compiling in a MetadataContainer.

Supported SOAP API Calls

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

Supported REST API HTTP Methods

Query, GET, POST, PATCH, DELETE

Fields

Field Name	Details
FullName	<p>Type string</p>

Field Name	Details
	<p>Properties Group, Nillable</p> <p>Description The full name of the associated object in the Metadata API. Use to avoid race conditions on create, before you have IDs. Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
Body	<p>Type string</p> <p>Properties Create, Update</p> <p>Description The data for the Apex class. The Body field is the only field you can update () or PATCH.</p>
Content	<p>Type string</p> <p>Properties None</p> <p>Description A string representation of ApexClassMetadata that lists the version, status, and packaged versions of the corresponding Apex class.</p>
ContentEntityId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description A reference to an Apex class. There can be only one ContentEntityId per ApexClassMember, otherwise, an error is reported. This field is required.</p>
LastSyncDate	<p>Type dateTime</p> <p>Properties Filter, Sort</p>

Field Name	Details
	<p>Description</p> <p>The date and time that this ApexClassMember Body was replicated from the underlying Apex class.</p> <p>When you deploy a MetadataContainer, this value is compared with the LastModifiedDate of the underlying Apex class. If LastSyncDate is older than LastModifiedDate, the deployment fails with an error.</p>
Metadata	<p>Type</p> <p>ApexClassMetadata</p> <p>Properties</p> <p>None</p> <p>Description</p> <p>An object that describes the version, status, and packaged versions of the corresponding Apex class.</p> <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
MetadataContainerId	<p>Type</p> <p>reference</p> <p>Properties</p> <p>Create, Filter, Group, Sort</p> <p>Description</p> <p>A reference to a MetadataContainer or ContainerAsyncRequest object.</p> <p>As part of a successful deployment, this field is reset from the ID of the deployed MetadataContainer to the ID of the corresponding ContainerAsyncRequest object.</p> <p>This field is required.</p>
SymbolTable	<p>Type</p> <p>SymbolTable</p> <p>Properties</p> <p>Nullable</p> <p>Description</p> <p>A complex type that represents all user-defined tokens in the Body of an ApexClass, ApexClassMember, or ApexTriggerMemeber and their associated line and column locations within the Body.</p> <p>This field is null if the symbol table cannot be created. A symbol table can't be created if the content referenced by the ContentEntityId field doesn't use a symbol table. Compiler errors for the last deployment of the MetadataContainer in the MetadataContainerId field also prevent a symbol table from being created.</p>

Usage

To edit, save, or compile an Apex class, create an ApexClassMember object that references it.



Note: Once an ApexClassMember is successfully deployed in a [MetadataContainer](#), the MetadataContainerId is changed to the ID of the [ContainerAsyncRequest](#), and the ApexClassMember can't be modified or reused.

Apex classes are often dependent on each other for functionality. For example, a method in one class can call a method in another class. If source file A is dependent on modified source file B and you try to save and compile source file A before you've saved the changes to source file B, the compiler will throw an error. To successfully save and compile a group of related source files, put the corresponding ApexClassMember and ApexTriggerMember objects in a single MetadataContainer object.

Each ApexClassMember object can only refer to a single MetadataContainer object. Multiple ApexClassMember objects can refer to the same MetadataContainer object.

ApexCodeCoverage

Represents code coverage test results for an Apex class or trigger. Available in Tooling API version 29.0 and later.

Supported SOAP API Calls

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

Supported REST API HTTP Methods

Query, GET

Fields

Field	Details
ApexTestClassId	Type string Properties Filter, Group, Sort Description The ID of the test class.
TestMethodName	Type string Properties Filter, Group, Sort Description The name of the test method.

Field	Details
ApexClassorTriggerId	Type string Properties Filter, Group, Sort Description The ID of the class or trigger under test.
NumLinesCovered	Type int Properties Filter, Group, Sort Description The number of covered lines.
NumLinesUncovered	Type int Properties Filter, Group, Sort Description The number of uncovered lines.
Coverage	Type complexvalue Properties None Description <p>Two lists of integers. The first is the covered lines, and the second is the list of uncovered lines. If a lines is missing from both lists, the line is not executable and does not require coverage.</p> <p>Coverage includes the following fields:</p> <ul style="list-style-type: none"> coveredLines namespace uncoveredLines

Usage


To query for code coverage, specify an Apex class, test class, or both. The returned JSON or XML object will contain two lists of integers: one for covered and one for uncovered lines.

The following example SOQL query retrieves code coverage results for a specific class or trigger covered by a specific test class:

```
SELECT Coverage
FROM ApexCodeCoverage
WHERE ApexClassOrTrigger = '01pD000000066GR'
AND ApexTestClass = '01pD000000064pu'
```

For per-class code coverage, the query would be:

```
SELECT Coverage
FROM ApexCodeCoverage
WHERE ApexClassOrTrigger = '01pD000000066GR'
```

 **Note:** In this case, multiple rows may be returned, since there may be multiple test classes that cover the same test class.

As noted above, `Coverage` is returned as two lists of integers. The first is the covered lines, and the second is the list of uncovered lines. If a line is missing from both lists, the line is not executable and does not require coverage. For example, if the covered lines are 2, 9, and 11, and uncovered lines are 3, 4, 5, and 6; the result would be: `{ 2, 9, 11 }, { 3, 4, 5, 6 }`. The missing lines (1, 7, 8 and 10) are not executable.

Code coverage percentage is a simple calculation of the number of covered lines divided by the sum of the number of covered lines and the number of uncovered lines. For example, to calculate code coverage percentage in SOAP:

```
ApexCodeCoverage acc = null; //Query for an ApexCodeCoverage object
Coverage coverage = acc.coverage;
int[] covered = coverage.coveredLines;
int[] uncovered = coverage.uncoveredLines;
int percent = covered.length / (covered.length + uncovered.length);
System.out.println("Total class coverage is " + percent + "%.");
```

ApexCodeCoverageAggregate

Represents aggregate code coverage test results for an Apex class or trigger. Available in Tooling API version 29.0 and later.

Supported SOAP API Calls

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

Supported REST API HTTP Methods

`Query`, `GET`, `DELETE`

Fields

Field	Details
<code>ApexClassOrTriggerId</code>	Type string
	Properties Filter, Group, Sort

Field	Details
	Description The ID of the class or trigger under test.
NumLinesCovered	Type int Properties Filter, Group, Sort Description The number of covered lines.
NumLinesUncovered	Type int Properties Filter, Group, Sort Description The number of uncovered lines.
Coverage	Type complexvalue Properties None Description Two lists of integers. The first is the covered lines, and the second is the list of uncovered lines. If a lines is missing from both lists, the line is not executable and does not require coverage. Coverage includes the following fields: <ul style="list-style-type: none"> coveredLines namespace uncoveredLines

Usage

To query for aggregate code coverage, specify an Apex test class. The returned JSON or XML object will contain two lists of integers: one for covered and one for uncovered lines. For examples, see [ApexCodeCoverage](#).

ApexComponent

Represents the saved copy of a Visualforce component. ApexComponent uses the cached version of the class unless one is unavailable. Available from API version 28.0 or later.

To edit, save, or compile Visualforce components, use [ApexComponentMember](#).

Supported SOAP API Calls

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

Supported REST API HTTP Methods

Query, GET, POST, PATCH, DELETE

Usage

To retrieve information about a Visualforce component, create an `ApexComponent` object that references it. For example code, see [Use Tooling API with SOAP](#).

To edit, save, or compile Visualforce components, use [ApexComponentMember](#).

ApexComponentMember

Represents the working copy of a Visualforce component for editing, saving or compiling in a `MetadataContainer`.

Supported SOAP API Calls

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

Supported REST API HTTP Methods

Query, GET, POST, PATCH, DELETE

Fields

Field Name	Details
FullName	<p>Type string</p> <p>Properties Group, Nillable</p> <p>Description The full name of the associated object in the Metadata API. Use to avoid race conditions on create, before you have IDs.</p> <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
Body	<p>Type string</p>

Field Name	Details
	<p>Properties Create, Update</p> <p>Description The data for the Visualforce component. The <code>Body</code> field is the only field you can <code>update ()</code> or <code>PATCH</code>.</p>
Content	<p>Type string</p> <p>Properties None</p> <p>Description A string representation of <code>ApexComponentMetadata</code> that lists the version, status, and packaged versions of the corresponding Visualforce component.</p>
ContentEntityId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description A reference to a Visualforce component. There can be only one <code>ContentEntityId</code> per <code>ApexComponentMember</code>, otherwise, an error is reported. This field is required.</p>
LastSyncDate	<p>Type dateTime</p> <p>Properties Filter, Sort</p> <p>Description The date that this <code>ApexComponentMember Body</code> was replicated from the underlying entity. When you deploy a <code>MetadataContainer</code>, this value is compared with the <code>LastModifiedDate</code> of the underlying Visualforce component. If <code>LastSyncDate</code> is older than <code>LastModifiedDate</code>, the deployment fails with an error.</p>
Metadata	<p>Type <code>ApexComponentMetadata</code></p> <p>Properties None</p>

Field Name	Details
	<p>Description</p> <p>An object that describes the version, status, and packaged versions of the corresponding Visualforce component.</p> <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
MetadataContainerId	<p>Type</p> <p>reference</p> <p>Properties</p> <p>Create, Filter, Group, Sort</p> <p>Description</p> <p>A reference to a MetadataContainer or ContainerAsyncRequest object.</p> <p>As part of a successful deployment, this field is reset from the ID of the deployed MetadataContainer to the ID of the corresponding ContainerAsyncRequest object.</p> <p>This field is required.</p>

Usage

To edit, save, or compile a Visualforce component, create an ApexComponentMember object that references it. To create a new Visualforce component, use the Force.com REST API or the Metadata API.



Note: Once an ApexComponentMember is successfully deployed in a [MetadataContainer](#), the `MetadataContainerId` is changed to the ID of the [ContainerAsyncRequest](#), and the ApexComponentMember can't be modified or reused.

Visualforce pages and components are often dependent on each other for functionality. To successfully save and compile a group of related source files, put the corresponding ApexComponentMember and ApexPageMember objects in a single MetadataContainer object.

Each ApexComponentMember object can only refer to a single MetadataContainer object. Multiple ApexComponentMember objects can refer to the same MetadataContainer object.

ApexExecutionOverlayAction

Specifies an Apex code snippet or SOQL query to execute at a specific line of code in an Apex class or trigger and optionally generate a heap dump.

Supported SOAP Calls

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

Supported REST HTTP Methods

Query, GET, POST, PATCH, DELETE

Fields

Field Name	Details
ActionScript	<p>Type string</p> <p>Properties Create, Nillable, Update</p> <p>Description The Apex code or SOQL query to run when execution reaches the line in the Apex class or trigger at the specified iteration. Results will be included in the heap dump file.</p>
ActionScriptType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Indicates whether the <code>ActionScript</code> is written in Apex or SOQL. Valid values are:</p> <ul style="list-style-type: none"> • None • Apex • SOQL <p>This field is required.</p>
ExecutableEntityId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort,</p> <p>Description The ID of the Apex class or trigger being executed. This field is required.</p>
ExpirationDate	<p>Type dateTime</p> <p>Properties Create, Filter, Sort, Update</p> <p>Description The expiration date of the overlay action. This field is required.</p>
IsDumpingHeap	<p>Type boolean</p> <p>Properties Create, Defaulted on create, Filter, Group, Sort, Update</p>

Field Name	Details
	Description Indicates whether a heap dump is generated (<code>true</code>) or not (<code>false</code>). To execute the <code>ActionScript</code> without generating a heap dump, set this field to <code>false</code> . This field is required.
Iteration	Type int Properties Create, Filter, Group, Sort, Update Description The number of times the specified line should execute before the heap dump is generated. This field is required.
Line	Type int Properties Create, Filter, Group, Sort, Update Description The line number of the heap dump marker. This field is required.
ScopeId	Type reference Properties Create, Filter, Group, Sort, Update Description The user who executed the action. This field is required.

Usage

When you are troubleshooting a runtime issue, you often want to find out more about the state of a variable or the state of the database, or create a specific condition to test your code. Use `ApexExecutionOverlayAction` to overlay a diagnostic output on an Apex class or trigger without compromising production code.

ApexExecutionOverlayResult

Represents the result from the Apex code snippet or SOQL query defined in the associated `ApexExecutionOverlayAction`, and the resulting heap dump if one was returned. Available from API version 28.0 or later.

Supported SOAP Calls

`query()`, `retrieve()`, `delete()`

Supported REST HTTP Methods

Query, GET, DELETE

Fields

Field Name	Details
ActionScript	<p>Type string</p> <p>Properties Nillable</p> <p>Description The Apex code or SOQL query that was run.</p>
ActionScriptType	<p>Type picklist</p> <p>Properties Filter, Group, Sort, Nillable</p> <p>Description Indicates whether the <code>ActionScript</code> is written in Apex or SOQL. Valid values are:</p> <ul style="list-style-type: none"> • None • Apex • SOQL
ApexResult	<p>Type ApexResult</p> <p>Properties Nillable</p> <p>Description A complex type that represents the result of Apex code executed as part of an <code>ApexExecutionOverlayAction</code>, returned in an <code>ApexExecutionOverlayResult</code>.</p>
ExpirationDate	<p>Type dateTime</p> <p>Properties Filter, Sort</p> <p>Description The expiration date of the overlay action.</p>
HeapDump	<p>Type HeapDump</p>

Field Name	Details
	<p>Properties Nillable</p> <p>Description A complex type that represents a heap dump in an ApexExecutionOverlayResult object. You can only have a single row when using HeapDump in SOQL. To select only one row, you can use a LIMIT=1 clause in your SOQL query, or you can list multiple rows for the user and have them select the row to inspect.</p>
IsDumpingHeap	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether a heap dump was generated (<code>true</code>) or not (<code>false</code>).</p>
Iteration	<p>Type int</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The number of times the specified line should execute before the heap dump is generated. This field is required.</p>
Line	<p>Type int</p> <p>Properties Filter, Group, Sort, Nillable</p> <p>Description The line number of the checkpoint.</p>
SOQLResult	<p>Type SOQLResult</p> <p>Properties Nillable</p> <p>Description A complex type that represents the result of a SOQL query in an ApexExecutionOverlayResult object.</p>
UserId	<p>Type reference</p> <p>Properties Filter, Group, Sort,</p>

Field Name	Details
	Description The user who executed the action.

Usage

When you are troubleshooting a runtime issue, you often want to find out more about the state of a variable or the state of the database, or create a specific condition to test your code. Use [ApexExecutionOverlayAction](#) to overlay a diagnostic output on an Apex class or trigger without compromising production code, and use [ApexExecutionOverlayResult](#) to navigate the results.

ApexLog

Represents a debug log.

To retrieve a raw log by ID, use the REST resource: `/subjects/ApexLog/id/Body/`. (Available from API version 28.0 or later.)

Supported SOAP Calls

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

Supported REST HTTP Methods

Query, GET, DELETE

Fields

Field	Details
Application	Type textarea Properties Filter, Group, Sort Description This value depends on the client type that triggered the log or heap dump. <ul style="list-style-type: none"> For API clients, this value is the client ID. For browser clients, this value is <code>Browser</code>. This field is required.
DurationMilliseconds	Type int Properties Filter, Group, Sort

Field	Details
	Description The duration of the transaction in milliseconds. This field is required.
Location	Type picklist Properties Filter, Group, Sort, Nillable, Restricted picklist Description Specifies the location of the origin of the log or heap dump. Values are: <ul style="list-style-type: none"> • Monitoring — Generated as part of debug log monitoring and visible to all administrators. These types of logs are maintained until the user or the system overwrites them. • SystemLog — Generated as part of system log monitoring and visible only to you. These types of logs are only maintained for 60 minutes or until the user clears them. • Preserved — A system log that is maintained longer than 60 minutes. Used for internal support.
LogLength	Type int Properties Filter, Group, Sort Description Length of the log or heap dump in bytes. This field is required.
LogUserId	Type reference Properties Filter, Group, Sort, Nillable Description ID of the user whose actions triggered the debug log or heap dump.
Operation	Type string Properties Filter, Group, Sort Description Name of the operation that triggered the debug log or heap dump, such as <code>APEXSOAP</code> , <code>Apex Sharing Recalculation</code> , and so on. This field is required.
Request	Type string

Field	Details
	<p>Properties Filter, Group, Sort</p> <p>Description Request type. Values are:</p> <ul style="list-style-type: none"> • <code>API</code> — Request came from an API. • <code>Application</code> — Request came from the Salesforce user interface. <p>This field is required.</p>
<code>StartTime</code>	<p>Type <code>dateTime</code></p> <p>Properties Filter, Sort</p> <p>Description Start time of the transaction. This field is required.</p>
<code>Status</code>	<p>Type <code>string</code></p> <p>Properties Filter, Group, Sort</p> <p>Description Status of the transaction. This value is either <code>Success</code>, or the text of an unhandled Apex exception. This field is required.</p>

ApexOrgWideCoverage

Represents code coverage test results for an entire organization. Available in Tooling API version 29.0 and later.

Supported SOAP API Calls

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

Supported REST API HTTP Methods

`Query`, `GET`, `DELETE`

Fields

Field	Details
PercentCovered	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The percentage of the code in the organization that is covered by tests.</p>

ApexPage

Represents the saved copy of an Apex page. ApexPage uses the cached version of the class unless one is unavailable. Available from API version 28.0 or later.

To edit, save, or compile Apex pages, use [ApexPageMember](#).

Supported SOAP API Calls

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

Supported REST API HTTP Methods

Query, GET, POST, PATCH, DELETE

Usage

To retrieve information about a Visualforce page, create an ApexPage object that references it. For example code, see [Use Tooling API with SOAP](#).

To edit, save, or compile Visualforce pages, use [ApexPageMember](#).

ApexPageMember

Represents the working copy of a Visualforce page for editing, saving or compiling in a MetadataContainer.

Supported SOAP API Calls

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

Supported REST API HTTP Methods

Query, GET, POST, PATCH, DELETE

Fields

Field Name	Details
FullName	<p>Type string</p> <p>Properties Group, Nillable</p> <p>Description The full name of the associated object in the Metadata API. Use to avoid race conditions on create, before you have IDs.</p> <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
Body	<p>Type string</p> <p>Properties Create, Update</p> <p>Description The data for the Visualforce page.</p> <p>The Body field is the only field you can update () or PATCH.</p>
Content	<p>Type string</p> <p>Properties None</p> <p>Description A string representation of ApexPageMetadata that lists the version, status, and packaged versions of the corresponding Visualforce page.</p>
ContentEntityId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description A reference to a Visualforce page.</p> <p>There can be only one ContentEntityId per ApexPageMember, otherwise, an error is reported.</p> <p>This field is required.</p>
LastSyncDate	<p>Type dateTime</p>

Field Name	Details
	<p>Properties Filter, Sort</p> <p>Description The date that this ApexPageMember Body was replicated from the underlying entity.</p> <p>When you deploy a MetadataContainer, this value is compared with the LastModifiedDate of the underlying Visualforce page. If LastSyncDate is older than LastModifiedDate, the deployment fails with an error.</p>
Metadata	<p>Type ApexPageMetadata</p> <p>Properties None</p> <p>Description An object that describes the version, status, and packaged versions of the corresponding Visualforce page.</p> <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
MetadataContainerId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description A reference to a MetadataContainer or ContainerAsyncRequest object.</p> <p>As part of a successful deployment, this field is reset from the ID of the deployed MetadataContainer to the ID of the corresponding ContainerAsyncRequest object.</p> <p>This field is required.</p>

Usage

To edit, save, or compile a Visualforce page, create an ApexPageMember object that references it. To create a new Visualforce page, use the Force.com REST API or the Metadata API.



Note: Once an ApexPageMember is successfully deployed in a [MetadataContainer](#), the MetadataContainerId is changed to the ID of the [ContainerAsyncRequest](#), and the ApexPageMember can't be modified or reused.


Visualforce pages and components are often dependent on each other for functionality. To successfully save and compile a group of related source files, put the corresponding ApexPageMember and ApexComponentMember objects in a single MetadataContainer object. Use ContainerAsyncRequest to send the MetadataContainer to the application server.

Each ApexPageMember object can only refer to a single MetadataContainer object. Multiple ApexPageMember objects can refer to the same MetadataContainer object.

ApexResult

A complex type that represents the result of Apex code executed as part of an `ApexExecutionOverlayAction`, returned in an `ApexExecutionOverlayResult`. Available from API version 28.0 or later.

Fields

Field	Details
<code>apexError</code>	<p>Type</p> <p>string</p> <p>Description</p> <p>The error text returned if the execution was unsuccessful.</p>
<code>apexExecutionResult</code>	<p>Type</p> <p><code>ExecuteAnonymousResult</code></p> <p>Description</p> <p>The structured result returned from a successful execution.</p> <p><code>ExecuteAnonymousResult</code> includes the following fields:</p> <ul style="list-style-type: none">• <code>column</code>• <code>compileProblem</code>• <code>compiled</code>• <code>exceptionMessage</code>• <code>exceptionStackTrace</code>• <code>line</code>• <code>success</code> <p> Note: <code>ExecuteAnonymousResult</code> is outside the current execution context and does not provide access to variables in the heap.</p>

Usage

Overlay Apex on checkpoints to capture structured debugging information. If your SOQL query may return more than one record when dealing with complex types, select only one row. For example, you can use a `LIMIT=1` clause in your SOQL query, or you can list rows for the user and have them select the row to inspect.

ApexTestQueueItem

Represents a single Apex class in the Apex job queue. Available from API version 30.0 or later.

Supported SOAP API Calls

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

Supported REST API HTTP Methods

Query, GET, POST, PATCH

Fields

Field Name	Details
ApexClassId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The Apex class whose tests are to be executed. This field can't be updated.</p>
Status	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The status of the test. Valid values are:</p> <ul style="list-style-type: none"> Queued Processing Aborted Completed Failed Preparing Holding <p>To abort a class that is in the Apex job queue, perform an update operation on the <code>ApexTestQueueItem</code> object and set its <code>Status</code> field to <code>Aborted</code>.</p>
ExtendedStatus	<p>Type string</p> <p>Properties Filter, Sort, Nillable</p> <p>Description The pass rate of the test run.</p>

Field Name	Details
	<p>For example: "(4/6)". This means that four out of a total of six tests passed.</p> <p>If the class fails to execute, this field contains the cause of the failure.</p>
ParentJobId	<p>Type reference</p> <p>Properties Filter, Group, Sort, Nillable,</p> <p>Description Read-only. Points to the <code>AsyncApexJob</code> that represents the entire test run.</p> <p>If you insert multiple Apex test queue items in a single bulk operation, the queue items will share the same parent job. This means that a test run can consist of the execution of the tests of several classes if all the test queue items are inserted in the same bulk operation.</p>

Usage

Insert an `ApexTestQueueItem` object to place its corresponding Apex class in the Apex job queue for execution. The Apex job executes the test methods in the class.

The example `RunTestListener.java` class below subscribes to the `TestResult` system topic and prints out the test results using `ApexTestQueueItem` and `ApexTestResult`. The example assumes the following:

- You have already set up a Java client application for Streaming API. This example uses the `org.cometd.client.BayeuxClient` created in the Java Client code example in the [Streaming API Developer's Guide](#).
- You have logged in `com.sforce.soap.tooling.SoapConnection`. For examples, see the [SOAP API Developer's Guide](#).



Note: The `RunTestListener.java` class must be instantiated after the Streaming API handshake. For example:

```
SoapConnection toolingConn = //Already set and logged in;
BayeuxClient client = //Already set and logged in;

//Listen on the handshake event
boolean handshaken = client.waitFor(10 * 1000, BayeuxClient.State.CONNECTED);
if (!handshaken) {
    System.out.println("Failed to handshake: " + client);
    System.exit(1);
}

final RunTestListener = null;
client.getChannel(Channel.META_SUBSCRIBE).addListener(
    new ClientSessionChannel.MessageListener() {
        public void onMessage(ClientSessionChannel channel, Message message) {
            boolean success = message.isSuccessful();
            if (success) {
                //Replace with your own ApexClass ids
```

```

        String apexTestClassId1 = "01pD00000007M0CIAU";
        String apexTestClassId2 = "01pD00000007NqtIAE";
        listener.runTests(new String[]{apexTestClassId1, apexTestClassId2});
    }
}
};
);
//This will subscribe to the TestRun system topic
listener = new RunTestListener(client, toolingConn);

```

```

import java.util.HashMap;
import org.cometd.bayeux.Message;
import org.cometd.bayeux.client.ClientSessionChannel;
import org.cometd.bayeux.client.ClientSessionChannel.MessageListener;
import org.cometd.client.BayeuxClient;

import com.sforce.soap.tooling.ApexTestQueueItem;
import com.sforce.soap.tooling.ApexTestResult;
import com.sforce.soap.tooling.QueryResult;
import com.sforce.soap.tooling.SObject;
import com.sforce.soap.tooling.SoapConnection;
import com.sforce.ws.ConnectionException;

public class RunTestListener {
    private static final String CHANNEL = "/systemTopic/TestResult";
    private SoapConnection conn;

    public RunTestListener(BayeuxClient client, SoapConnection conn) {
        this.conn = conn;
        System.out.println("Subscribing for channel: " + CHANNEL);
        client.getChannel(CHANNEL).subscribe(new MessageListener() {
            @Override
            public void onMessage(ClientSessionChannel channel, Message message) {
                HashMap data = (HashMap) message.getData();
                HashMap subject = (HashMap) data.get("subject");
                String id = (String) subject.get("Id");
                System.out.println("\nAysncApexJob " + id);
                getTestQueueItems(id);
            }
        });
    }

    public void runTests(String[] apexTestClassIds) {
        if (apexTestClassIds.length == 0) {
            System.out.println("No test to run");
            return;
        }
        System.out.println("Running async test run");
        String ids = apexTestClassIds[0];
        for (int i = 1; i < apexTestClassIds.length; i++) {
            ids += "," + apexTestClassIds[i];
        }
    }
}

```

```

    try {
        conn.runTestsAsynchronous(ids);
    } catch (ConnectionException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

private void getTestQueueItems(String asyncApexJobId) {
    try {
        QueryResult res = conn
            .query("SELECT Id, Status, ApexClassId FROM ApexTestQueueItem WHERE ParentJobId
= '"
                + asyncApexJobId + "'");
        if (res.getSize() > 0) {
            for (SObject o : res.getRecords()) {
                ApexTestQueueItem atqi = (ApexTestQueueItem) o;
                System.out.println("\tApexTestQueueItem - "+atqi.getStatus());
                if (atqi.getStatus().equals("Completed")) {
                    getApexTestResults(atqi.getId());
                }
            }
        } else {
            System.out.println("No queued items for " + asyncApexJobId);
        }
    } catch (ConnectionException e) {
        e.printStackTrace();
    }
}

private void getApexTestResults(String apexTestQueueItemId) {
    try {
        QueryResult res = conn
            .query("SELECT StackTrace,Message, AsyncApexJobId,MethodName, Outcome,ApexClassId
FROM ApexTestResult WHERE QueueItemId = '"
                + apexTestQueueItemId + "'");
        if (res.getSize() > 0) {
            for (SObject o : res.getRecords()) {
                ApexTestResult atr = (ApexTestResult) o;
                System.out.println("\tTest result for "
                    + atr.getApexClassId() + "." + atr.getMethodName());
                String msg = atr.getOutcome().equals("Fail") ? " - "
                    + atr.getMessage() + " " + atr.getStackTrace() : "";
                System.out.println("\t\tTest " + atr.getOutcome() + msg);
            }
        } else {
            System.out.println("No Test Results for " + apexTestQueueItemId);
        }
    } catch (ConnectionException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}
}

```

ApexTestResult

Represents the result of an Apex test method execution. Available from API version 30.0 or later.

Supported SOAP API Calls

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

Supported REST API HTTP Methods

Query, GET

Fields

Field Name	Details
ApexClassId	Type reference Properties Filter, Group, Sort Description The Apex class whose test methods were executed.
ApexLogId	Type reference Properties Filter, Group, Nillable, Sort Description Points to the <code>ApexLog</code> for this test method execution if debug logging is enabled; otherwise, <code>null</code> .
AsyncApexJobId	Type reference Properties Filter, Group, Nillable, Sort Description Read-only. Points to the <code>AsyncApexJob</code> that represents the entire test run. This field points to the same object as <code>ApexTestQueueItem.ParentJobId</code> .
Message	Type string

Field Name	Details
	Properties Filter, Nillable, Sort Description The exception error message if a test failure occurs; otherwise, <code>null</code> .
MethodName	Type string Properties Filter, Group, Nillable, Sort Description The name of the test method.
Outcome	Type picklist Properties Filter, Group, Restricted picklist, Sort Description The result of the test. Valid values are: <ul style="list-style-type: none"> • Pass • Failed • CompileFail • Skip
QueueItemId	Type reference Properties Filter, Group, Nillable, Sort Description Points to the ApexTestQueueItem which is the class that this test method is part of.
StackTrace	Type string Properties Filter, Nillable, Sort Description The Apex stack trace if the test failed; otherwise, <code>null</code> .
TestTimestamp	Type dateTime

Field Name	Details
	Properties Filter, Sort
	Description The start time of the test method.

Usage

You can query the fields of the ApexTestResult record that corresponds to a test method executed as part of an Apex class execution. Each test method execution is represented by a single ApexTestResult record. For example, if an Apex test class contains six test methods, six ApexTestResult records are created. These records are in addition to the ApexTestQueueItem record that represents the Apex class. For example code, see [ApexTestQueueItem](#).

ApexTrigger

Represents the saved copy of an Apex trigger. ApexTrigger uses the cached version of the class unless one is unavailable. Available from API version 28.0 or later.

To edit, save, or compile Apex triggers, use [ApexTriggerMember](#).

Supported SOAP API Calls

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


Supported REST API HTTP Methods

Query, GET, POST, PATCH, DELETE

Fields

Field Name	Details
ApiVersion	Type double
	Properties Create, Filter, Sort, Update
	Description The API version for this trigger. Every trigger has an API version specified at creation.
Body	Type string

Field Name	Details
	Properties Create, Nillable, Update Description The Apex trigger definition. Limit: 1 million characters.
BodyCrc	Type double Properties Create, Defaulted on create, Filter, Nillable, Sort, Update Description The CRC (cyclic redundancy check) of the class or trigger file.
EntityDefinitionId	Type string Properties Filter, Group, Nillable, Sort Description The Id of the EntityDefinition object associated with this object.
IsValid	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort, Update Description Indicates whether any dependent metadata has changed since the trigger was last compiled (<code>true</code>) or not (<code>false</code>).
LengthWithoutComments	Type int Properties Create, Filter, Group, Sort, Update Description Length of the trigger without comments.
Metadata	Type ApexTriggerMetadata Properties None Description An object that describes the version, status, and packaged versions of the corresponding Apex trigger.

Field Name	Details
	<p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
Status	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The current status of the Apex trigger. The following string values are valid:</p> <ul style="list-style-type: none"> • Active—The trigger is active. • Inactive—The trigger is inactive, but not deleted. • Deleted—The trigger is marked for deletion. This is useful for managed packages, because it allows a class to be deleted when a managed package is updated. <p> Note: Inactive is not valid for ApexClass. For more information, see the Force.com Metadata API Developer's Guide.</p>
UsageAfterDelete	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Specifies whether the trigger is an after delete trigger (true) or not (false).</p>
UsageAfterInsert	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Specifies whether the trigger is an after insert trigger (true) or not (false).</p>
UsageAfterUndelete	<p>Type boolean</p> <p>Properties Create, Filter, Update</p> <p>Description Specifies whether the trigger is an after undelete trigger (true) or not (false).</p>
UsageAfterUpdate	<p>Type boolean</p>

Field Name	Details
	Properties Create, Filter, Update Description Specifies whether the trigger is an after update trigger (<code>true</code>) or not (<code>false</code>).
UsageBeforeDelete	Type boolean Properties Create, Filter, Update Description Specifies whether the trigger is an before delete trigger (<code>true</code>) or not (<code>false</code>).
UsageBeforeInsert	Type boolean Properties Create, Filter, Update Description Specifies whether the trigger is an before insert trigger (<code>true</code>) or not (<code>false</code>).
UsageBeforeUpdate	Type boolean Properties Create, Filter, Update Description Specifies whether the trigger is an before update trigger (<code>true</code>) or not (<code>false</code>).
UsageIsBulk	Type boolean Properties Create, Filter, Update Description Specifies whether the trigger is defined as a bulk trigger (<code>true</code>) or not (<code>false</code>).

Usage

To retrieve information about an Apex trigger, create an ApexTrigger object that references it. For example code, see [Use Tooling API with SOAP](#).

To edit, save, or compile Apex triggers, use [ApexTriggerMember](#).

ApexTriggerMember

Represents the working copy of an Apex trigger for editing, saving or compiling in a MetadataContainer.

Supported SOAP API Calls

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

Supported REST API HTTP Methods

Query, GET, POST, PATCH, DELETE

Fields

Field Name	Details
FullName	<p>Type string</p> <p>Properties Group, Nillable</p> <p>Description The full name of the associated object in the Metadata API. Use to avoid race conditions on create, before you have IDs.</p> <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
Body	<p>Type string</p> <p>Properties Create, Update</p> <p>Description The data for the Apex trigger.</p> <p>The Body field is the only field you can <code>update()</code> or <code>PATCH</code>.</p>
Content	<p>Type string</p> <p>Properties None</p> <p>Description A string representation of ApexTriggerMetadata that lists the version, status, and packaged versions of the corresponding Apex trigger.</p>

Field Name	Details
ContentEntityId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description A reference to an Apex trigger. There can be only one ContentEntityId per ApexTriggerMember, otherwise, an error is reported. This field is required.</p>
LastSyncDate	<p>Type dateTime</p> <p>Properties Filter, Sort</p> <p>Description The date that this ApexTriggerMember Body was replicated from the underlying entity. When you deploy a MetadataContainer, this value is compared with the LastModifiedDate of the underlying Apex trigger. If LastSyncDate is older than LastModifiedDate, the deployment fails with an error.</p>
Metadata	<p>Type ApexTriggerMetadata</p> <p>Properties None</p> <p>Description An object that describes the version, status, and packaged versions of the corresponding Apex trigger. Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
MetadataContainerId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description A reference to a MetadataContainer or ContainerAsyncRequest object. As part of a successful deployment, this field is reset from the ID of the deployed MetadataContainer to the ID of the corresponding ContainerAsyncRequest object. This field is required.</p>

Field Name	Details
SymbolTable	<p>Type SymbolTable</p> <p>Properties Nillable</p> <p>Description A complex type that represents all user-defined tokens in the <code>Body</code> of an <code>ApexClass</code>, <code>ApexClassMember</code>, or <code>ApexTriggerMemeber</code> and their associated line and column locations within the <code>Body</code>.</p> <p>This field is null if the symbol table cannot be created. A symbol table can't be created if the content referenced by the <code>ContentEntityId</code> field doesn't use a symbol table. Compiler errors for the last deployment of the MetadataContainer in the <code>MetadataContainerId</code> field also prevent a symbol table from being created.</p>

Usage

To edit, save, or compile an Apex trigger, create an `ApexTriggerMember` object that references it. To create a new trigger, use the Force.com REST API or the Metadata API.



Note: Once an `ApexTriggerMember` is successfully deployed in a [MetadataContainer](#), the `MetadataContainerId` is changed to the ID of the [ContainerAsyncRequest](#), and the `ApexTriggerMember` can't be modified or reused.

Apex triggers and classes are often dependent on each other for functionality. For example, a method in one class can call a method in another class. If source file A is dependent on modified source file B and you try to save and compile source file A before you've saved the changes to source file B, the compiler will throw an error. To successfully save and compile a group of related source files, put the corresponding `ApexTriggerMember` and `ApexClassMember` objects in a single `MetadataContainer` object. Use `ContainerAsyncRequest` to send the `MetadataContainer` to the application server.

Each `ApexTriggerMember` object can only refer to a single `MetadataContainer` object. Multiple `ApexTriggerMember` objects can refer to the same `MetadataContainer` object.

AuraDefinition

Represents a Lightning definition, such as component markup, a client-side controller, or an event. Lightning components is a beta feature. Available in API version 32.0 and later.

Supported SOAP API Calls

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

Supported REST API HTTP Methods

Query, GET, POST, PATCH, DELETE

Fields

Field Name	Details
AuraDefinitionBundleId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of the bundle containing the definition. A bundle contains a Lightning definition and all its related resources.</p>
DefType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The definition type, such as COMPONENT for component markup, or CONTROLLER for a client-side controller.</p>
Format	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The format of the definition. Valid values are:</p> <ul style="list-style-type: none">• XML for component markup• JS for JavaScript code• CSS for styles
Source	<p>Type textarea</p> <p>Properties Create, Update</p> <p>Description The contents of the Lightning definition. This is all the markup or code for the definition.</p>

Usage

For more information, see the [Lightning Components Developer's Guide](#).

AuraDefinitionBundle

Represents a Lightning definition bundle, such as a component or application bundle. A bundle contains a Lightning definition and all its related resources. Lightning components is a beta feature. Available in API version 32.0 and later.

Supported SOAP API Calls


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

Supported REST API HTTP Methods

Query, GET, POST, PATCH, DELETE

Fields

Field Name	Details
ApiVersion	Type double Properties Create, Filter, Sort, Update Description The API version for this bundle. Every bundle has an API version specified at creation.
Description	Type textarea Properties Create, Filter, Group, Sort, Update Description The text description of the bundle. Maximum size of 255 characters.
DeveloperName	Type string Properties Create, Filter, Group, Sort, Update Description The unique name of the record in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your organization. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. This field is automatically generated but you can supply your own value if you create the record using the API.

Field Name	Details
	 Note: When creating large sets of data, always specify a unique <code>DeveloperName</code> for each record. If no <code>DeveloperName</code> is specified, performance may slow while Salesforce generates one for each record.
Language	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The language of the <code>MasterLabel</code>.</p>
MasterLabel	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description Master label for the Lightning bundle. This internal label doesn't get translated.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the <code>namespacePrefix__componentName</code> notation.</p> <p>The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are not Developer Edition organizations, <code>NamespacePrefix</code> is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.

Usage

For more information, see the [Lightning Components Developer's Guide](#).

BusinessProcess

Represents a business process.

This object is available in API version 33.0 and later.

Supported SOAP Calls

`create()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

Supported REST HTTP Methods

GET, PATCH, POST

Fields

Field	Details
Description	Type string Properties Create, Filter, Group, Nillable, Sort, Update Description The business process description, limited to 255 characters.
IsActive	Type boolean Properties Defaulted on create, Filter, Group, Sort, Update Description Indicates whether this business process is active (<code>true</code>) or not (<code>false</code>).
Name	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description The process name.
NamespacePrefix	Type string Properties Filter, Group, Nillable, Sort

Field	Details
	Description A unique string to distinguish this type from any others.

CompactLayout

Represents the values that define a compact page layout.

This object is available in API version 32.0 and later.

Supported SOAP Calls

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

Supported REST HTTP Methods

DELETE, GET, PATCH, POST

Fields

Field	Details
DeveloperName	Type string Properties Filter, Group, Sort Description The developer's internal name for the compact layout (for example, "CL_c") used in the API.
FullName	Type string Properties Create, Group, Nillable Description The unique name used as the compact layout identifier for API access. The <code>FullName</code> can contain only underscores and alphanumeric characters. It must be unique, begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.
Metadata	Type mns:CompactLayout

Field	Details
	<p>Properties Create, Nillable, Update</p> <p>Description The compact layout metadata.</p> <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace of the package of which the compact layout is a part.</p>
ObjectType	<p>Type Restricted picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The type of object used in the layout, such as an Account or Lead.</p>

CompactLayoutItemInfo

Represents a field selected for a compact layout, and the order of that field in the compact layout.

This object is available in API version 32.0 and later.

Supported SOAP Calls

`query()`

Supported REST HTTP Methods

GET

Fields

Field	Details
CompactLayoutInfo	<p>Type CompactLayoutInfo</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The compact layout associated with this CompactLayoutItemInfo.</p>
CompactLayoutInfoId	<p>Type Id</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the compact layout associated with this field.</p>
DurableId	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description This field reserved for future use. Do not use.</p>
FieldDefinition	<p>Type FieldDefinition</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Required. The field definition for this field.</p>
FieldDefinitionId	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Required. ID of this field.</p>
SortOrder	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field	Details
	Description The order of the field in the compact layout. 1 is first.

SOQL Limitations

This object doesn't support some SOQL operations.

COUNT()

Example Query: `SELECT COUNT() FROM EntityDefinition`

Error Returned: The "DurableId" field is of type number, but the value from the external data source is "Account.SearchResult". Contact your administrator for help.

GROUP BY

Example Query: `SELECT COUNT(qualifiedapiname), isfeedenabled FROM EntityDefinition GROUP BY isfeedenabled`

Error Returned: The requested operation is not yet supported by this SObject storage type, contact salesforce.com support for more information.

LIMIT, LIMIT OFFSET

Example Queries:

`SELECT qualifiedapiname FROM EntityDefinition LIMIT 5`

`SELECT qualifiedapiname FROM EntityDefinition LIMIT 5 OFFSET 10`

An incorrect result is returned because LIMIT and LIMIT OFFSET are ignored.

NOT

Example Query: `SELECT qualifiedapiname FROM EntityDefinition WHERE qualifiedapiname!='Account'`

Error Returned: Only equals comparisons permitted

OR

Example Query: `SELECT qualifiedapiname, keyprefix FROM EntityDefinition WHERE isdeletable=true OR (isfeedenabled=false AND keyprefix='01j')`

Error Returned: Disjunctions not supported

CompactLayoutInfo

Represents the metadata for a custom or standard compact layout.

This object is available in API version 32.0 and later.

Supported SOAP Calls

`query()`

Supported REST HTTP Methods

GET

Fields

Field	Details
DeveloperName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The developer's internal name for the compact layout (for example, <code>CL_c</code>) used in the API.</p>
DurableId	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description Unique identifier for the field. Always retrieve this value before using it, as the value isn't guaranteed to stay the same from one release to the next. Use this field to simplify queries.</p>
EntityDefinition	<p>Type EntityDefinition</p> <p>Properties Filter, Group, Sort</p> <p>Description Required. Available starting with version 32.0. The entity definition for the object associated with this CompactLayoutInfo.</p>
EntityDefinitionId	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Required. ID of the record associated with this CompactLayoutInfo. The record's object type is in <code>EntityDefinition</code>.</p>
FullName	<p>Type string</p> <p>Properties Filter, Group, Sort</p>

Field	Details
	<p>Description</p> <p>The unique name used as the compact layout identifier for API access. The <code>fullName</code> can contain only underscores and alphanumeric characters. It must be unique, begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores.</p> <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
IsDefault	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Defaulted on create, Filter, Group, Sort</p> <p>Description</p> <p>If <code>true</code>, this compact layout is the default for the associated object.</p>
Items	<p>Type</p> <p>QueryResult</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>A foreign key field pointing to CompactLayoutItemsInfo. Use in subqueries.</p>
Label	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Nillable, Sort</p> <p>Description</p> <p>The compact layout's label.</p>
Metadata	<p>Type</p> <p>mns:CompactLayout</p> <p>Properties</p> <p>Create, Nillable, Update</p> <p>Description</p> <p>Metadata that defines compact layouts.</p> <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
NamespacePrefix	<p>Type</p> <p>string</p>

Field	Details
	<p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the <i>namespacePrefix__componentName</i> notation.</p> <p>The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are not Developer Edition organizations, NamespacePrefix is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.

SOQL Limitations

This object doesn't support some SOQL operations.

COUNT()

Example Query: `SELECT COUNT() FROM EntityDefinition`

Error Returned: The "DurableId" field is of type number, but the value from the external data source is "Account.SearchResult". Contact your administrator for help.

GROUP BY

Example Query: `SELECT COUNT(qualifiedapiname), isfeedenabled FROM EntityDefinition GROUP BY isfeedenabled`

Error Returned: The requested operation is not yet supported by this SObject storage type, contact salesforce.com support for more information.

LIMIT, LIMIT OFFSET

Example Queries:

`SELECT qualifiedapiname FROM EntityDefinition LIMIT 5`

`SELECT qualifiedapiname FROM EntityDefinition LIMIT 5 OFFSET 10`

An incorrect result is returned because LIMIT and LIMIT OFFSET are ignored.

NOT

Example Query: `SELECT qualifiedapiname FROM EntityDefinition WHERE qualifiedapiname!='Account'`

Error Returned: Only equals comparisons permitted

OR

Example Query: `SELECT qualifiedapiname, keyprefix FROM EntityDefinition WHERE isdeletable=true OR (isfeedenabled=false AND keyprefix='01j')`

Error Returned: Disjunctions not supported

ContainerAsyncRequest

Allows you to compile and asynchronously deploy a MetadataContainer object to your organization.

Supported SOAP API Calls


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

Supported REST API HTTP Methods

Query, GET, POST

Fields

Field Name	Details
DeployDetails	Type DeployDetails Properties Nillable Description Provides detailed XML for any compile errors reported during an asynchronous request. Includes <code>componentFailures</code> . Replaces the JSON field <code>CompilerErrors</code> in Tooling API version 31.0 and later.
ErrorMsg	Type textarea Properties Nillable Description Errors reported during an asynchronous request.
IsCheckOnly	Type boolean Properties Create, Defaulted on create, Filter, Group, Sort Description Indicates whether the asynchronous request compiles the code without making any changes to the organization (<code>true</code>) or compiles and saves the code (<code>false</code>). This field is required.

Field Name	Details
	 Note: You can compile without saving but you can't save without compiling.
IsRunTests	<p>Type boolean</p> <p>Properties None</p> <p>Description Reserved for future use.</p>
MetadataContainerId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description The ID of a MetadataContainer object. Specify a <code>MetadataContainerId</code> or a <code>MetadataContainerMemberId</code>, but not both.</p>
MetadataContainerMemberId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description The ID of an ApexClassMember, ApexTriggerMember, ApexPageMember or ApexComponentMember object. Specify a <code>MetadataContainerId</code> or a <code>MetadataContainerMemberId</code>, but not both.</p>
State	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The state of the request. Valid values are:</p> <ul style="list-style-type: none"> • <code>Queued</code>—the job is in the queue. • <code>Invalidated</code>—Salesforce cancelled the job because the results might not be valid. This state occurs if someone changes the container members while <code>IsCheckOnly=true</code>, or if a newer compile request is added to the queue. • <code>Completed</code>—the compilation or deployment finished. The <code>SymbolTable</code> fields for the specified object(s) were successfully updated. If

Field Name	Details
	<p>If <code>IsCheckOnly</code> is <code>false</code>, the <code>Body</code> for each object was saved and the <code>MetadataContainerId</code> field for each object was reset from the ID of the deployed MetadataContainer to the ID of the corresponding ContainerAsyncRequest object.</p> <ul style="list-style-type: none"> Failed—the compilation or deployment failed for the reasons stated in the <code>CompilerError</code> field. Error—an unexpected error occurred. The messages in the <code>ErrorMsg</code> field can be provided to Salesforce support if the issue persists. Aborted—use this value to delete a queued deployment. <p>This field is required.</p>

Usage

When you deploy a `ContainerAsyncRequest`, you must specify whether to save the compiled entities:

- To compile entities without saving, set the request to `IsCheckOnly=true`. This option is only supported if a `MetadataContainerMember` is specified. A single `MetadataContainerMemberId` can't be compiled without saving.
- To compile and save entities to your organization, set the request to `IsCheckOnly=false`.

If the compile succeeds, the `SymbolTable` field is updated on each object in the specified `MetadataContainer`. If the save or compile fails and a `SymbolTable` field cannot be updated, the field is cleared. If there is an outstanding save request, all updates, inserts, and deployments fail.

To terminate a queued deployment, set the `State` field to `Aborted`.

CustomField

Represents a custom field on a custom object that stores data unique to your organization. Includes access to the associated `CustomField` object and related fields in Salesforce Metadata API. Available from API version 28.0 or later.

Supported SOAP Calls

`create()`, `query()`, `retrieve()`, `search()`, `update()`, `upsert()`

Supported REST HTTP Methods

`Query`, `GET`, `POST`, `PATCH`

Fields

Field Name	Details
<code>DeveloperName</code>	Type string

Field Name	Details
	<p>Properties Filter, Group, Sort</p> <p>Description The developer's internal name for the custom field (for example "CF_c").</p>
Metadata	<p>Type CustomFieldMetadata</p> <p>Properties Create, Nillable, Update</p> <p>Description CustomFieldMetadata includes the following fields:</p> <ul style="list-style-type: none"> • caseSensitive • customDataType • defaultValue • deleteConstraint • deprecated • description • displayFormat • displayLocationInDecimal • escapeMarkup • externalDeveloperName • externalId • formula • formulaTreatBlanksAs • inlineHelpText • isFilteringDisabled • isNameField • isSortingDisabled • label • length • maskChar • maskType • picklist • populateExistingRows • precision • readOnlyProxy • referenceTo • relationshipLabel • relationshipName

Field Name	Details
	<ul style="list-style-type: none"> relationshipOrder reparentableMasterDetail required restrictedAdminField scale startingNumber stripMarkup summarizedField summaryFilterItems summaryForeignKey summaryOperation trackFeedHistory trackHistory type unique visibleLines writeRequiresMasterRead <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace of the custom field. A custom field can be in an extension namespace different than the object.</p>
TableEnumOrId	<p>Type Restricted picklist</p> <p>Properties Filter, Group, Sort</p> <p>Description The enum (for example, Account) or ID of the object this field is on.</p>

CustomFieldMember

Represents the working copy of a field for editing or saving in a MetadataContainer. This object is available in API version 33.0 and later.

Supported SOAP API Calls

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

Supported REST API HTTP Methods

Query, GET, POST, PATCH, DELETE

Fields

Field Name	Details
Content	<p>Type string</p> <p>Properties None</p> <p>Description A string representation of CustomField that contains the field's metadata.</p>
ContentEntityId	<p>Type ID</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description A reference to a custom field. There can be only one <code>ContentEntityId</code> per CustomField, otherwise, an error is reported.</p>
FullName	<p>Type string</p> <p>Properties Group, Nillable</p> <p>Description The full name of the associated object in the Metadata API. Use to avoid race conditions on create, before you have IDs. Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
IsDeleted	<p>Type boolean</p> <p>Properties Group, Nillable</p>

Field Name	Details
	Description Indicates whether the object is marked as deleted (<code>true</code>) or not (<code>false</code>).
LastSyncDate	Type dateTime Properties Filter, Sort Description The date that this CustomField was replicated from the underlying entity.
Metadata	Type CustomField Properties None Description An object that describes the version, status, and packaged versions of the corresponding CustomField. Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.

CustomObject

Represents a custom object that stores data unique to your organization. Includes access to the associated CustomObject object and related fields in Salesforce Metadata API. Available from API version 31.0 or later.

Supported SOAP Calls

`query()`, `retrieve()`, `search()`

Supported REST HTTP Methods

Query, GET

Fields

Field Name	Details
CustomHelpId	Type ID

Field Name	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description The control that contains the help content if this custom object has customized help content.</p>
Description	<p>Type string</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The object's description. This can be useful to describe the reason for creating the object or its intended use.</p>
DeveloperName	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The developer's internal name for the custom object (for example "CF_c").</p>
ExternalName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Maps to a table in the external data source. If you created the external object using Validate and Sync for the data source, this name is automatically created.</p>
ExternalRepository	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Maps to a table in the external data source. If you created the external object using Validate and Sync for the data source, this name is automatically created; do not modify it.</p>
Language	<p>Type string</p> <p>Properties Filter, Group, Restricted picklist, Sort</p>

Field Name	Details
	<p>Description</p> <p>The language of the action. Valid values are:</p> <ul style="list-style-type: none"> Chinese (Simplified): zh_CN Chinese (Traditional): zh_TW Danish: da Dutch: nl_NL English: en_US Finnish: fi French: fr German: de Italian: it Japanese: ja Korean: ko Norwegian: no Portuguese (Brazil): pt_BR Russian: ru Spanish: es Spanish (Mexico): es_MX Swedish: sv Thai: th
NamespacePrefix	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Nillable, Sort</p> <p>Description</p> <p>The namespace of the package of which the custom object is a part.</p>
SharingModel	<p>Type</p> <p>picklist</p> <p>Properties</p> <p>Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description</p> <p>The sharing model. Values are:</p> <ul style="list-style-type: none"> Edit ControlledByparent None Read

CustomTab

Represents a custom tab.

This object is available in the Tooling API version 33.0 and later.

Supported Calls

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

Fields

Field	Details
ContentId	Type reference Properties Filter, Group, Nillable, Sort Description Read-only. The ID of the item that the custom tab points to. For Lightning components, this is the ID of the component bundle. For custom object tabs, this field is <code>null</code> .
Description	Type string Properties Filter, Nillable, Sort Description The tab's description.
DeveloperName	Type string Properties Filter, Group, Nillable, Sort Description The developer's internal name for the custom tab.
EncodingKey	Type string Properties Filter, Group, Nillable, Sort Description Read-only. Type of encoding assigned to the URL called by the tab. The default encoding setting is Unicode: UTF-8. Change it if you are passing information to a URL that requires

Field	Details
	<p>data in a different format. This option is available when the value <code>URL</code> is selected in the tab type. Valid values are:</p> <ul style="list-style-type: none"> • <code>UTF-8</code>—Unicode (UTF-8) • <code>ISO-8859-1</code>—General US & Western Europe (ISO-8859-1, ISO-LATIN-1) • <code>Shift_JIS</code>—Japanese (Shift-JIS) • <code>ISO-2022-JP</code>—Japanese (JIS) • <code>EUC-JP</code>—Japanese (EUC-JP) • <code>x-SJIS_0213</code>—Japanese (Shift-JIS_2004) • <code>ks_c_5601-1987</code>—Korean (ks_c_5601-1987) • <code>Big5</code>—Traditional Chinese (Big5) • <code>GB2312</code>—Simplified Chinese (GB2312) • <code>Big5-HKSCS</code>—Traditional Chinese Hong Kong (Big5-HKSCS)
FullName	<p>Type string</p> <p>Properties Create, Group, Nillable</p> <p>Description The name of the tab. The value of this field depends on the type of tab, and the API version.</p> <ul style="list-style-type: none"> • For custom object tabs, the <code>fullName</code> is the developer-assigned name of the custom object (<code>MyCustomObject__c</code>, for example). • For Web tabs, the <code>fullName</code> is the developer-assigned name of the tab (<code>MyWebTab</code>, for example). <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
HasSidebar	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates if the tab displays the sidebar panel.</p>
MasterLabel	<p>Type string</p> <p>Properties Filter, Group, idLookup, Nillable, Sort</p> <p>Description Required. The label for the custom tab, which displays in Setup.</p>

Field	Details
Metadata	<p>Type CustomTabMetadata</p> <p>Properties Create, Nillable, Update</p> <p>Description Custom tab metadata.</p> <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
MotifName	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description Read-only. The name of the tab style assigned to the custom tab.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace of the package of which the custom tab is a part.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The type of custom tab. Valid values are:</p> <ul style="list-style-type: none"> • apexPage • aura • customObject • flexiPage • sControl • url
Url	<p>Type string</p> <p>Properties Filter, Nillable, Sort</p>

Field	Details
	Description The URL for the external web-page to embed in this tab.

DataType

Represents the datatype of a field. Use this object with EntityDefinition, EntityParticle, or FieldDefinition to simplify queries. Available in Tooling API version 34.0 and later.

Fields

Field	Details
DurableId	Type string Properties Filter, Group, Nillable, Sort Description Unique identifier for the field. Always retrieve this value before using it, as the value isn't guaranteed to stay the same from one release to the next. Use this field to simplify queries.
Fields	Type QueryResult Properties Filter, Group, Sort Description Fields of the specified datatype on the object specified in a query.
IsComplex	Type boolean Properties Defaulted on create, Filter, Group, Sort Description If <code>true</code> , the datatype contains other datatypes, in contrast to a simple datatype like string.
Name	Type string Properties Filter, Group, Nillable, Sort Description The name of the datatype, such as <code>string</code> or <code>CustomField</code> .

Field	Details
Namespace	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace, if any, for the datatype.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description <p>The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the <i>namespacePrefix__componentName</i> notation.</p> <p>The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. In organizations that are not Developer Edition organizations, NamespacePrefix is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects. </p>

Example

From an object, retrieve all the fields of one datatype.

```
SELECT DataType, QualifiedApiName
FROM EntityParticle
WHERE DataType = 'phone' AND
      EntityDefinition.QualifiedApiName = 'Account'
```

SOQL Limitations

This object doesn't support some SOQL operations.

COUNT()

Example Query: `SELECT COUNT() FROM EntityDefinition`

Error Returned: The "DurableId" field is of type number, but the value from the external data source is "Account.SearchResult". Contact your administrator for help.

GROUP BY

Example Query: `SELECT COUNT(qualifiedapiname), isfeedenabled FROM EntityDefinition GROUP BY isfeedenabled`

Error Returned: The requested operation is not yet supported by this SObject storage type, contact salesforce.com support for more information.

LIMIT, LIMIT OFFSET

Example Queries:

`SELECT qualifiedapiname FROM EntityDefinition LIMIT 5`

`SELECT qualifiedapiname FROM EntityDefinition LIMIT 5 OFFSET 10`

An incorrect result is returned because LIMIT and LIMIT OFFSET are ignored.

NOT

Example Query: `SELECT qualifiedapiname FROM EntityDefinition WHERE qualifiedapiname!='Account'`

Error Returned: Only equals comparisons permitted

OR

Example Query: `SELECT qualifiedapiname, keyprefix FROM EntityDefinition WHERE isdeletable=true OR (isfeedenabled=false AND keyprefix='01j')`

Error Returned: Disjunctions not supported

DeployDetails

A complex type that contains detailed XML for any compile errors reported in the asynchronous request defined by a ContainerAsyncRequest object. Replaces the JSON field `CompilerErrors` in Tooling API version 31.0 and later.

Fields

Field	Details
componentFailures	<p>Type</p> <p>string</p> <p>Description</p> <p>The line number, component name and a short description for any compile errors. For example:</p> <pre><DeployDetails> <componentFailures> <lineNumber>5</lineNumber> <fullName>myApex</fullName> <problem>invalid name 'abc'</problem> </componentFailures> <componentFailures> <lineNumber>10</lineNumber> <fullName>myApex2</fullName> <problem>invalid type 'hello'</problem> </componentFailures> </DeployDetails></pre>

Field	Details
	<pre></componentFailures> </DeployDetails></pre>

EmailTemplate

Represents an email template.

This object is available in API version 32.0 and later.

Supported SOAP Calls

`create()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`

Supported REST HTTP Methods

DELETE, GET, PATCH, POST

Fields

Field	Details
ApiVersion	Type double Properties Filter, Nillable, Sort Description The API version if this is a Visualforce email template. Every Visualforce email template has an API version specified at creation.
Description	Type string Properties Filter, Nillable, Sort Description The email template description. This can be useful to describe the reason for creating the template or its intended use.
FullName	Type string Properties Create, Group, Nillable

Field	Details
	<p>Description</p> <p>The unique name used as the template identifier for API access. The <code>fullName</code> can contain only underscores and alphanumeric characters. It must be unique, begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores.</p> <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
Metadata	<p>Type</p> <p>EmailTemplateMetadata</p> <p>Properties</p> <p>Create, Nillable, Update</p> <p>Description</p> <p>Email template metadata.</p> <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
Name	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, idLookup, Sort</p> <p>Description</p> <p>The email template name.</p>
NamespacePrefix	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Nillable, Sort</p> <p>Description</p> <p>A unique string to distinguish this template from any others. For example, if this template is being using by a flow, use the <code>NamespacePrefix</code> to uniquely identify the templates in multiple flow instances.</p>
Subject	<p>Type</p> <p>string</p> <p>Properties</p> <p>Group, Nillable, Sort</p> <p>Description</p> <p>The email subject.</p>

EntityDefinition

Provides row-based access to metadata about standard and custom objects.

This object is available in API version 32.0 and later.

Supported SOAP Calls

`query()`

Supported REST HTTP Methods

GET

Fields

Field	Details
ApexTriggers	<p>Type QueryResult</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Represents the Apex triggers associated with this object. Available in Tooling API starting version 34.0.</p>
AssignmentRules	<p>Type QueryResult</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Represents assignment rules that allow you to automatically route cases to the appropriate users or queues. Available in Tooling API version 34.0 and later.</p>
AutoResponseRules	<p>Type QueryResult</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Represents the auto-response rules defined for the object. Available in Tooling API starting version 34.0.</p>
BusinessProcesses	<p>Type QueryResult</p>

Field	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description Represents the business processes defined for the object. Business processes display different picklist values for users based on their profile and associated record type. Available in Tooling API starting version 34.0.</p>
ChildRelationships	<p>Type QueryResult</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Represents the child relationships defined for the object. Available in Tooling API starting version 34.0.</p>
CompactLayouts	<p>Type QueryResult</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Represents the compact layouts defined for the object. Available in Tooling API starting version 34.0.</p>
CustomFields	<p>Type QueryResult</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Represents the custom fields defined for the object. Available in Tooling API starting version 34.0.</p>
DefaultCompactLayout	<p>Type CompactLayoutInfo</p> <p>Properties Create, Nillable, Update</p> <p>Description Metadata about the compact layout defined as the default for this object, if any.</p>
DefaultCompactLayoutId	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field	Details
	Description ID of the default compact layout, if any.
DetailUrl	Type string Properties Filter, Group, Nillable, Sort Description URL to the read-only detail page for this object. Corresponds to the <code>urlDetail</code> field in <code>DescribeSobjectResult</code> . Available in Tooling API starting version 34.0.
DeveloperName	Type string Properties Filter, Group, Nillable, Sort Description The developer's internal name for the custom object (for example "CF_c").
DurableId	Type string Properties Filter, Group, Nillable, Sort Description Unique identifier for the field. Always retrieve this value before using it, as the value isn't guaranteed to stay the same from one release to the next. Use this field to simplify queries.
EditDefinitionUrl	Type string Properties Filter, Group, Nillable, Sort Description Available in Tooling API starting version 34.0.
EditUrl	Type string Properties Filter, Group, Nillable, Sort Description The URL used when editing the custom entity definition. Corresponds to the <code>urlEdit</code> field on <code>DescribeSobjectResult</code> . Available in Tooling API starting version 34.0.

Field	Details
FieldSets	<p>Type QueryResult</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Represents the field sets defined for the object.</p>
Fields	<p>Type QueryResult</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Represents the standard and custom fields defined for this object.</p>
FullName	<p>Type string</p> <p>Properties Create, Group, Nillable</p> <p>Description The name of the entity. If a field, the name must specify the parent object, for example <code>Account.FirstName</code>. Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
HelpSettingPageName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The name of the custom help setting page. Available in Tooling API starting version 34.0.</p>
HelpSettingPageUrl	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The URL pointing to the custom help setting page for custom objects. Available in Tooling API starting version 34.0.</p>
IsApexTriggerable	<p>Type boolean</p>

Field	Details
	<p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, Apex triggers can be defined for the entity.</p>
<code>IsCompactLayoutable</code>	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, the object supports compact layouts. That is, compact layouts can be defined, a system compact layout can be synthesized, or both.</p>
<code>IsCreatable</code>	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, records based on the object (<code>true</code>) can be created.</p>
<code>IsCustomizable</code>	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, custom fields can be defined for the entity.</p>
<code>IsDeletable</code>	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, the object (<code>true</code>) can be deleted.</p>
<code>IsFeedEnabled</code>	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, the Chatter feed is enabled for this object. Available in Tooling API starting version 34.0.</p>

Field	Details
IsQueryable	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, the object can be queried.</p>
IsWorkflowEnabled	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, workflow rules can be defined for the entity.</p>
KeyPrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The first three digits of the entity's ID, which identify the object type, such as Account or Opportunity.</p>
Label	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The label for this object, used in the compact layout and in the user's language locale.</p>
Layouts	<p>Type QueryResult</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Represents the layouts defined for this object. Available in Tooling API starting version 34.0.</p>
Limits	<p>Type QueryResult</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field	Details
	<p>Description</p> <p>The limits defined for this object. Corresponds to the Limits page for each standard object in Setup, or the Limits related list for each custom object. Available in Tooling API starting version 34.0.</p>
LookupFilters	<p>Type</p> <p>QueryResult</p> <p>Properties</p> <p>Filter, Group, Nillable, Sort</p> <p>Description</p> <p>Represents the lookup filters defined for this object. Available in Tooling API starting version 34.0.</p>
MasterLabel	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>The label for this object, which displays in Setup. The master label is in the default language locale for the organization. If there is no default language locale, the label is in <code>en_US</code>.</p>
Metadata	<p>Type</p> <p><code>mns : CustomObject</code></p> <p>Properties</p> <p>Create, Nillable, Update</p> <p>Description</p> <p>Metadata about the standard or custom object. Details are available in the CustomObject entry in the metadata namespace in the Tooling API WSDL.</p> <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
NamespacePrefix	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Nillable, Sort</p> <p>Description</p> <p>The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the <code>namespacePrefix__componentName</code> notation.</p> <p>The namespace prefix can have one of the following values:</p>

Field	Details
	<ul style="list-style-type: none"> In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. In organizations that are not Developer Edition organizations, <code>NamespacePrefix</code> is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.
Particles	<p>Type QueryResult</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The particles defined for this object. Available in Tooling API starting version 34.0.</p>
PluralLabel	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The plural version of the object's <code>Label</code>.</p>
Publisher	<p>Type Publisher</p> <p>Properties Create, Nillable, Update</p> <p>Description The publisher of this object, for example Salesforce, a user, or a package name. Available in Tooling API starting version 34.0.</p>
PublisherId	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the publisher associated with this object. Available in Tooling API starting version 34.0.</p>
QualifiedApiName	<p>Type string</p> <p>Properties Filter, Group, Sort</p>

Field	Details
	<p>Description</p> <p>A unique external ID for the entity of the form <code>NamespacePrefix__DeveloperName</code> for standard objects and <code>NamespacePrefix__DeveloperName__c</code> for custom objects.</p>
QuickActionDefinitions	<p>Type</p> <p>QueryResult</p> <p>Properties</p> <p>Filter, Group, Nillable, Sort</p> <p>Description</p> <p>Represents the quick actions defined for this object. Available in Tooling API starting version 34.0.</p>
RecordTypes	<p>Type</p> <p>QueryResult</p> <p>Properties</p> <p>Filter, Group, Nillable, Sort</p> <p>Description</p> <p>Represents the record types defined for this object. Available in Tooling API starting version 34.0.</p>
RelationshipDomains	<p>Type</p> <p>QueryResult</p> <p>Properties</p> <p>Filter, Group, Nillable, Sort</p> <p>Description</p> <p>Metadata about the relationships with other objects that this object has. Available in Tooling API starting version 34.0.</p>
RunningUserEntityAccess	<p>Type</p> <p>UserEntityAccess</p> <p>Properties</p> <p>Create, Nillable, Update</p> <p>Description</p> <p>Represents the running user's access to this object. Available in Tooling API starting version 34.0.</p>
RunningUserEntityAccessId	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Nillable, Sort</p> <p>Description</p> <p>ID of the UserEntityAccess record associated with this object. Available in Tooling API starting version 34.0.</p>

Field	Details
SearchLayouts	<p>Type QueryResult</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Represents the search layouts associated with this object. Available in Tooling API starting version 34.0.</p>
StandardActions	<p>Type QueryResult</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Represents the standard actions defined for this object. Available in Tooling API starting version 34.0.</p>
ValidationRules	<p>Type QueryResult</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Represents the validation rules defined for this object. Available in Tooling API starting version 34.0.</p>
WebLinks	<p>Type QueryResult</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Represents the Weblinks associated with this object. Available in Tooling API starting version 34.0.</p>
WorkflowAlerts	<p>Type QueryResult</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Represents the workflow alerts associated with this object. Available in Tooling API starting version 34.0.</p>
WorkflowFieldUpdates	<p>Type QueryResult</p>

Field	Details
	Properties Filter, Group, Nillable, Sort Description Represents the workflow field updates for this object. Available in Tooling API starting version 34.0.
WorkflowOutboundMessages	Type QueryResult Properties Filter, Group, Nillable, Sort Description Represents the workflow outbound messages associated with this object. Available in Tooling API starting version 34.0.
WorkflowTasks	Type QueryResult Properties Filter, Group, Nillable, Sort Description Represents the workflow tasks associated with this object. Available in Tooling API starting version 34.0.

SOQL Limitations

This object doesn't support some SOQL operations.

COUNT()

Example Query: `SELECT COUNT() FROM EntityDefinition`

Error Returned: The "DurableId" field is of type number, but the value from the external data source is "Account.SearchResult". Contact your administrator for help.

GROUP BY

Example Query: `SELECT COUNT(qualifiedapiname), isfeedenabled FROM EntityDefinition GROUP BY isfeedenabled`

Error Returned: The requested operation is not yet supported by this SObject storage type, contact salesforce.com support for more information.

LIMIT, LIMIT OFFSET

Example Queries:

`SELECT qualifiedapiname FROM EntityDefinition LIMIT 5`

`SELECT qualifiedapiname FROM EntityDefinition LIMIT 5 OFFSET 10`

An incorrect result is returned because LIMIT and LIMIT OFFSET are ignored.

NOT

Example Query: `SELECT qualifiedapiname FROM EntityDefinition WHERE qualifiedapiname!='Account'`

Error Returned: Only equals comparisons permitted

OR

Example Query: `SELECT qualifiedapiname, keyprefix FROM EntityDefinition WHERE isdeletable=true OR (isfeedenabled=false AND keyprefix='01j')`

Error Returned: Disjunctions not supported

EntityLimit

Represents the limits for an object as displayed in the Setup UI.

This object is available in API version 34.0 and later.

Supported SOAP Calls

`query()`

Supported REST HTTP Methods

GET

Fields

Field	Details
DurableId	Type string Properties Filter, Group, Nillable, Sort Description Unique identifier for the field. Always retrieve this value before using it, as the value isn't guaranteed to stay the same from one release to the next. Use this field to simplify queries.
EntityDefinition	Type EntityDefinition Properties Filter, Group, Sort Description The object to which these limits apply.
EntityDefinitionId	Type string

Field	Details
	<p>Properties Filter, Group, Sort</p> <p>Description ID of the object to which these limits apply.</p>
Label	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The label of the object to which these limits apply.</p>
Max	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The maximum number of objects that the organization is allowed to have.</p>
Remaining	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of objects still available. For example, if the limit on custom objects is 100, and you create 75, this value is 25.</p>
Type	<p>Type string</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description What type of component the limit applies to:</p> <ul style="list-style-type: none"> • ActiveLookupFilters • ActiveRules • ActiveValidationRules • ApprovalProcesses • CbsSharingRules • CustomFields • CustomRelationship • RollupSummary

Field	Details
	<ul style="list-style-type: none"> • SharingRules • TotalRules • VLookup

SOQL Limitations

This object doesn't support some SOQL operations.

COUNT()

Example Query: `SELECT COUNT() FROM EntityDefinition`

Error Returned: The "DurableId" field is of type number, but the value from the external data source is "Account.SearchResult". Contact your administrator for help.

GROUP BY

Example Query: `SELECT COUNT(qualifiedapiname), isfeedenabled FROM EntityDefinition GROUP BY isfeedenabled`

Error Returned: The requested operation is not yet supported by this SObject storage type, contact salesforce.com support for more information.

LIMIT, LIMIT OFFSET

Example Queries:

`SELECT qualifiedapiname FROM EntityDefinition LIMIT 5`

`SELECT qualifiedapiname FROM EntityDefinition LIMIT 5 OFFSET 10`

An incorrect result is returned because LIMIT and LIMIT OFFSET are ignored.

NOT

Example Query: `SELECT qualifiedapiname FROM EntityDefinition WHERE qualifiedapiname!='Account'`

Error Returned: Only equals comparisons permitted

OR

Example Query: `SELECT qualifiedapiname, keyprefix FROM EntityDefinition WHERE isdeletable=true OR (isfeedenabled=false AND keyprefix='01j')`

Error Returned: Disjunctions not supported

EntityParticle

Represents each element of a field that can be presented in a user interface. Contrast EntityParticle with FieldDefinition, which represents each element of a field defined in the Metadata API. Available in Tooling API version 34.0 and later.

Supported SOAP Calls

`query()`

Supported REST HTTP Methods

GET

Fields

Field	Details
<code>DataType</code>	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description Data type of the field, for example <code>Text (40)</code> or <code>Date/Time</code>. The values are defined as they are in the user interface, not the corresponding API data type names. Available in Tooling API starting version 34.0.</p>
<code>DeveloperName</code>	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description Required. The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your organization. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Label is Record Type Name.</p>
<code>DurableId</code>	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description Unique identifier for the field. Always retrieve this value before using it, as the value isn't guaranteed to stay the same from one release to the next. Use this field to simplify queries.</p>
<code>EntityDefinition</code>	<p>Type EntityDefinition</p> <p>Properties Filter, Group, Sort</p>

Field	Details
	<p>Description</p> <p>A relationship lookup to the object type that contains this field. For example, if the field is defined on an account, the lookup is to <code>Account</code>. You can't interact directly with this field. Instead, use it in queries.</p> <pre>SELECT EntityDefinition.Label FROM EntityParticle WHERE EntityParticle.QualifiedApi Name= 'Lead '</pre>
EntityDefinitionId	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>The durable ID for the object defined in the <code>EntityDefinition</code> field.</p>
ExtraTypeInfo	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>Represents further definition of a type. Available in Tooling API starting version 34.0.</p> <ul style="list-style-type: none"> For type textarea: <ul style="list-style-type: none"> plaintextarea richtextarea For type URL, image For type reference, <ul style="list-style-type: none"> externallookup indirectlookup externallookup For Account, switchablepersonname personname
IsApiFilterable	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Defaulted on create, Filter, Group, Sort</p> <p>Description</p> <p>If <code>true</code>, the field can be specified in the <code>WHERE</code> clause of a query string. Available in Tooling API starting version 34.0.</p>

Field	Details
IsApiGroupable	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, the field can be included in the GROUP BY clause of a SOQL query. Available in Tooling API starting version 34.0.</p>
IsApiSortable	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, a query can sort on this field. Available in Tooling API starting version 34.0.</p>
IsCalculated	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, the field's value is calculated. Available in Tooling API starting version 34.0.</p>
IsCompactLayoutable	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, the field can be included in a compact layout. Available in Tooling API starting version 34.0.</p>
IsFieldHistoryTracked	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, the field's history can be tracked. Available in Tooling API starting version 34.0.</p>
IsHighScaleNumber	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p>

Field	Details
	<p>Description</p> <p>Indicates whether the field stores numbers to 8 decimal places regardless of what's specified in the field details (<code>true</code>) or not (<code>false</code>). Used to handle currencies for products that cost fractions of a cent and are sold by the thousands. If high-scale unit pricing isn't enabled in your organization, this field isn't returned. Available in Tooling API starting version 34.0.</p>
<code>IsListVisible</code>	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Defaulted on create, Filter, Group, Sort</p> <p>Description</p> <p>If <code>true</code>, the field can be included in a related list. Available in Tooling API starting version 34.0.</p>
<code>IsNameField</code>	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Defaulted on create, Filter, Group, Sort</p> <p>Description</p> <p>If <code>true</code>, the field is a name field. Available in Tooling API starting version 34.0.</p>
<code>IsNullible</code>	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Defaulted on create, Filter, Group, Sort</p> <p>Description</p> <p>If <code>true</code>, the field can be left out of queries on the object. Available in Tooling API starting version 34.0.</p>
<code>IsWorkflowFilterable</code>	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Defaulted on create, Filter, Group, Sort</p> <p>Description</p> <p>If <code>true</code>, the field can be filtered for a workflow. Available in Tooling API starting version 34.0.</p>
<code>Label</code>	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>The label that corresponds to the field in the user interface.</p>

Field	Details
Length	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The maximum number of bytes available to store the value in this field. Available in Tooling API starting version 34.0.</p>
MasterLabel	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description Master label for this object. This display value is the internal label that is not translated. Limit: 40 characters.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the <i>namespacePrefix__componentName</i> notation.</p> <p>The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are not Developer Edition organizations, NamespacePrefix is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.
Precision	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The maximum number of digits allowed for this field. Available in Tooling API starting version 34.0.</p>

Field	Details
QualifiedApiName	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description A unique external name for the field.</p>
Scale	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of digits to the right of the decimal in an integer. For example, 3.00 has a scale of 2. Available in Tooling API starting version 34.0.</p>
ServiceDataType	<p>Type DataType</p> <p>Properties Filter, Group, Sort</p> <p>Description The service datatype for this field. Available in Tooling API starting version 34.0.</p>
ServiceDataTypeId	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the ServiceDataType. Available in Tooling API starting version 34.0.</p>

Example Query

```
SELECT DataType, FieldDefinition.QualifiedApiName
FROM EntityParticle
WHERE EntityDefinition.QualifiedApiName ='Account'
```

SOQL Limitations

This object doesn't support some SOQL operations.

COUNT()

Example Query: `SELECT COUNT() FROM EntityDefinition`

Error Returned: The "DurableId" field is of type number, but the value from the external data source is "Account.SearchResult". Contact your administrator for help.

GROUP BY

Example Query: `SELECT COUNT(qualifiedapiname), isfeedenabled FROM EntityDefinition GROUP BY isfeedenabled`

Error Returned: The requested operation is not yet supported by this SObject storage type, contact salesforce.com support for more information.

LIMIT, LIMIT OFFSET

Example Queries:

`SELECT qualifiedapiname FROM EntityDefinition LIMIT 5`

`SELECT qualifiedapiname FROM EntityDefinition LIMIT 5 OFFSET 10`

An incorrect result is returned because LIMIT and LIMIT OFFSET are ignored.

NOT

Example Query: `SELECT qualifiedapiname FROM EntityDefinition WHERE qualifiedapiname!='Account'`

Error Returned: Only equals comparisons permitted

OR

Example Query: `SELECT qualifiedapiname, keyprefix FROM EntityDefinition WHERE isdeletable=true OR (isfeedenabled=false AND keyprefix='01j')`

Error Returned: Disjunctions not supported

FieldDefinition

Represents a standard or custom field, providing row-based access to field metadata. Contrast FieldDefinition with EntityParticle, which represents each element of a field that can be presented in a user interface.

This object is available in API version 32.0 and later.

Supported SOAP Calls

`query()`

Supported REST HTTP Methods

GET

Fields

Field	Details
CompactLayoutItems	Type QueryResult


Field	Details
	<p>Properties Filter, Group, Sort</p> <p>Description The CompactLayoutItemInfo records associated with this field definition. Available in Tooling API starting version 34.0. Use in subqueries, such as:</p> <pre>SELECT Id, QualifiedApiName, (SELECT DurableId, SortOrder FROM CompactLayoutItems) FROM FieldDefinition WHERE EntityDefinition.QualifiedApiName = 'Account' AND QualifiedApiName = 'Name'</pre>
ControlledFields	<p>Type QueryResult</p> <p>Properties Filter, Group, Sort</p> <p>Description The controlled fields in a dependent picklist. Use in subqueries to reduce the number of queries. Available in Tooling API starting version 34.0.</p>
ControllingFieldDefinition	<p>Type FieldDefinition</p> <p>Properties Filter, Group, Sort</p> <p>Description The field definition of the controlling field if this field is a dependent picklist. A dependent picklist works with a controlling picklist or checkbox to filter the available options. The value chosen in the controlling field affects the values available in the dependent field. This field is available in API version 14.0 and later. Available in Tooling API starting version 34.0.</p>
ControllingFieldDefinitionId	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the <code>ControllingFieldDefinition</code> for this field. Available in Tooling API starting version 34.0.</p>
DataType	<p>Type string</p> <p>Properties Filter, Group, Sort</p>

Field	Details
	<p>Description</p> <p>Data type of the field, for example <code>Text (40)</code> or <code>Date/Time</code>. The values are defined as they are in the user interface, not the corresponding API data type names. Available in Tooling API starting version 34.0.</p>
DeveloperName	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>Required. The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your organization. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Label is Record Type Name.</p>
DurableId	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>Unique identifier for the field. Always retrieve this value before using it, as the value isn't guaranteed to stay the same from one release to the next. Use this field to simplify queries.</p>
EntityDefinition	<p>Type</p> <p>EntityDefinition</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>A relationship lookup to the object type that contains this field. For example, if the field is defined on an account, the lookup is to <code>Account</code>. You can't interact directly with this field. Instead, use it in queries.</p> <pre>SELECT EntityDefinition.Label FROM FieldDefinition WHERE EntityDefinition.QualifiedApi Name= 'Lead '</pre>
EntityDefinitionId	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>The durable ID for the object defined in the <code>EntityDefinition</code> field.</p>

Field	Details
ExtraTypeInfo	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description Represents further definition of a type. Available in Tooling API starting version 34.0.</p> <ul style="list-style-type: none"> For type textarea: <ul style="list-style-type: none"> plaintextarea richtextarea For type URL, image For type reference, <ul style="list-style-type: none"> externallookup indirectlookup externallookup For Account, <ul style="list-style-type: none"> switchablepersonname personname
FullName	<p>Type string</p> <p>Properties Create, Group, Nillable</p> <p>Description The full name of the associated metadata object in Metadata API.</p> <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
IsApiFilterable	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, the field can be specified in the <code>WHERE</code> clause of a query string. Available in Tooling API starting version 34.0.</p>
IsApiGroupable	<p>Type boolean</p>

Field	Details
	<p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, the field can be included in the GROUP BY clause of a SOQL query. Available in Tooling API starting version 34.0.</p>
<code>IsApiSortable</code>	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, a query can sort on this field. Available in Tooling API starting version 34.0.</p>
<code>IsCalculated</code>	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, the field's value is calculated. Available in Tooling API starting version 34.0.</p>
<code>IsCompactLayoutable</code>	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, the field can be included in a compact layout. Available in Tooling API starting version 34.0.</p>
<code>IsFieldHistoryTracked</code>	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, the field's history can be tracked. Available in Tooling API starting version 34.0.</p>
<code>IsHighScaleNumber</code>	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the field stores numbers to 8 decimal places regardless of what's specified in the field details (<code>true</code>) or not (<code>false</code>). Used to handle currencies for products that cost</p>

Field	Details
	fractions of a cent and are sold by the thousands. If high-scale unit pricing isn't enabled in your organization, this field isn't returned. Available in Tooling API starting version 34.0.
IsHtmlFormatted	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, the field contains HTML. Available in Tooling API starting version 34.0.</p>
IsListFilterable	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, the field can be filtered for a related list. Available in Tooling API starting version 34.0.</p>
IsListSortable	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, the field can be sorted for a related list. Available in Tooling API starting version 34.0.</p>
IsListVisible	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, the field can be included in a related list. Available in Tooling API starting version 34.0.</p>
IsNameField	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, the field is a name field. Available in Tooling API starting version 34.0.</p>
IsNillable	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p>

Field	Details
	<p>Description</p> <p>If <code>true</code>, the field can be left out of queries on the object. Available in Tooling API starting version 34.0.</p>
IsWorkflowFilterable	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Defaulted on create, Filter, Group, Sort</p> <p>Description</p> <p>If <code>true</code>, the field can be filtered for a workflow. Available in Tooling API starting version 34.0.</p>
Label	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>The label that corresponds to the field in the user interface.</p>
Length	<p>Type</p> <p>int</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>The maximum number of bytes available to store the value in this field. Available in Tooling API starting version 34.0.</p>
LookupFilters	<p>Type</p> <p>QueryResult</p> <p>Properties</p> <p>Filter, Group, Nillable, Sort</p> <p>Description</p> <p>The lookup filters associated with the field. Use when writing subqueries.</p> <pre>SELECT DurableId, QualifiedApiName, (SELECT Id, SourceObject, SourceFieldDefinition.Label, IsOptional, Active, Developer Name, LastModifiedBy.Name, LastModifiedDate FROM LookupFilters) FROM EntityDefinition WHERE QualifiedApiName = 'User' "</pre> <p> Note: LookupFilter is not supported on the article type object.</p>
MasterLabel	<p>Type</p> <p>string</p>

Field	Details
	<p>Properties Filter, Group, Sort</p> <p>Description Master label for this object. This display value is the internal label that is not translated. Limit: 40 characters.</p>
Metadata	<p>Type CustomField</p> <p>Properties Create, Nillable, Update</p> <p>Description Compact layout metadata, from the <code>mns</code> namespace. Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the <code>namespacePrefix__componentName</code> notation. The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are not Developer Edition organizations, <code>NamespacePrefix</code> is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.
Particles	<p>Type QueryResult</p> <p>Properties Filter, Group, Sort</p> <p>Description The EntityParticles associated with this field. Available in Tooling API starting version 34.0.</p>

Field	Details
Precision	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The maximum number of digits allowed for this field. Available in Tooling API starting version 34.0.</p>
Publisher	<p>Type Publisher</p> <p>Properties Filter, Group, Sort</p> <p>Description The publisher of this field, for example Salesforce, a user, or a package name. Available in Tooling API starting version 34.0.</p>
PublisherId	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the publisher associated with this field. Available in Tooling API starting version 34.0.</p>
QualifiedApiName	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description A unique external name for the field.</p>
ReferenceTargetField	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description Available only for indirect lookup relationship fields on external objects. Specifies the field on the parent object to match against this indirect lookup relationship field, whose values come from an external data source. The specified custom field on the parent object must have both <code>externalId</code> and <code>unique</code> set to <code>true</code>. Available in Tooling API starting version 34.0.</p>
ReferenceTo	<p>Type RelationshipReferenceTo</p>

Field	Details
	<p>Properties Filter, Group, Sort</p> <p>Description Represents the reference this field has to another object. Available in Tooling API starting version 34.0.</p>
RelationshipDomains	<p>Type QueryResult</p> <p>Properties Filter, Group, Sort</p> <p>Description Metadata about the relationships with other objects that this field has. Available in Tooling API starting version 34.0.</p>
RelationshipName	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The value for one-to-many relationships. For example, in the object MyObject with a relationship to YourObject, the relationship name is typically YourObjects. Available in Tooling API starting version 34.0.</p>
RunningUserFieldAccessId	<p>Type string</p> <p>Properties</p> <p>Description Don't use this field. Available in Tooling API starting version 34.0.</p>
Scale	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of digits to the right of the decimal in an integer. For example, 3.00 has a scale of 2. Available in Tooling API starting version 34.0.</p>
ServiceDataType	<p>Type DataType</p> <p>Properties Filter, Group, Sort</p>

Field	Details
	Description The service datatype for this field. Available in Tooling API starting version 34.0.
ServiceDataTypeId	Type string Properties Filter, Group, Sort Description ID of the ServiceDataType. Available in Tooling API starting version 34.0.
ServiceDataTypes	Type QueryResult Properties Filter, Group, Sort Description The ServiceDataTypes associated with this field. Available in Tooling API starting version 34.0.
WorkflowFieldUpdates	Type QueryResult Properties Filter, Group, Sort Description The workflow field updates for this field. Field updates allow you to automatically update a field value to one that you specify when a workflow rule is triggered. Available in Tooling API starting version 34.0.

Example Query

```
SELECT QualifiedApiName, (Select DataType From Particles)
FROM FieldDefinition
WHERE EntityDefinition.QualifiedApiName ='Account'
```

CustomField Metadata

Field Name	Field Type	Description
caseSensitive	boolean	Indicates whether the field is case sensitive (<code>true</code>) or not (<code>false</code>). For indirect lookup relationship fields on external objects, this attribute affects how this custom field's values are matched against the values of <code>referenceTargetField</code> .
defaultValue	string	If specified, represents the default value of the field.

Field Name	Field Type	Description
<code>deleteConstraint</code>	DeleteConstraint is an enumeration of strings	<p>Provides deletion options for lookup relationships. Valid values are:</p> <p><code>SetNull</code> Default value.</p> <p>If the lookup record is deleted, the lookup field is cleared.</p> <p><code>Restrict</code> If the record is in a lookup relationship, prevents it from being deleted.</p> <p><code>Cascade</code> Deletes the lookup record and associated lookup fields.</p> <p>For more information on lookup relationships, see “Object Relationships” in the Salesforce Help.</p>
<code>description</code>	string	Description of the field.
<code>displayFormat</code>	string	The display format.
<code>displayLocationInDecimal</code>	boolean	Indicates how the value of a Geolocation custom field appears in the user interface. If <code>true</code> , the geolocation appears in decimal notation. If <code>false</code> , the geolocation appears as degrees, minutes, and seconds.
<code>externalDeveloperName</code>	string	Available only for external objects. Name of the table column on the external data source that maps to this custom field in Salesforce. Corresponds to <code>External Column Name</code> in the user interface. This field is available in API version 32.0 and later.
<code>externalId</code>	boolean	Indicates whether the field is an external ID field (<code>true</code>) or not (<code>false</code>).
<code>formula</code>	string	If specified, represents a formula on the field.
<code>formulaTreatBlankAs</code>	TreatBlanksAs	Indicates how to treat blanks in a formula. Valid values are <code>BlankAsBlank</code> or <code>BlankAsZero</code> .
<code>fullName</code>	string	<p>Required. The internal name of the object. White spaces and special characters are escaped for validity. The name must:</p> <ul style="list-style-type: none"> Contain characters, letters, or the underscore (<code>_</code>) character Must start with a letter Can't end with an underscore Can't contain two consecutive underscore characters.
<code>indexed</code>	boolean	Indicates if the field is indexed. If this field is unique or the <code>externalId</code> is set true, the <code>isIndexed</code> value is set to true. This field has been deprecated as of version 14.0 and is only provided for backward compatibility.
<code>inlineHelpText</code>	string	Represents the content of field-level help. For more information, see “Defining Field-Level Help” in the Salesforce Help.
<code>isFilteringDisabled</code>	boolean	Available only for external objects. Indicates whether the custom field is available in filters. This field is available in API version 32.0 and later.

Field Name	Field Type	Description
isNameField	boolean	Available only for external object fields of type text. For each external object, you can specify one field as the name field. If you set this field's value to <code>true</code> , make sure that the external table column identified by <code>externalDeveloperName</code> contains name values. This field is available in API version 32.0 and later.
isSortingDisabled	boolean	Available only for external objects. Indicates whether the custom field is sortable. This field is available in API version 32.0 and later.
reparentableMasterDetail	boolean	Indicates whether the child records in a master-detail relationship on a custom object can be reparented to different parent records. The default value is <code>false</code> . This field is available in API version 25.0 and later.
label	string	Label for the field. You cannot update the label for standard picklist fields, such as the <code>Industry</code> field for accounts.
length	int	Length of the field.
lookupFilter	LookupFilter	The Lookup filter definition for the custom field. Available in API version 30.0 and later.
maskChar	EncryptedFieldMaskChar	For encrypted fields, specifies the character to be used as a mask. Valid values are: <ul style="list-style-type: none"> • asterisk • x
maskType	EncryptedFieldMaskType	For encrypted text fields, specifies the format of the masked and unmasked characters in the field. Valid values: <p><code>all</code> All characters in the field are hidden. This option is equivalent to the <code>Mask All Characters</code>.</p> <p><code>creditCard</code> The first 12 characters are hidden, the last four are displayed. This option is equivalent to the <code>Credit Card Number</code> option.</p> <p><code>ssn</code> The first five characters are hidden, the last four are displayed. This option is equivalent to the <code>Social Security Number</code> option.</p> <p><code>lastFour</code> All characters are hidden, except the last four are displayed. This option is equivalent to the <code>Last Four Characters Clear</code> option.</p> <p><code>sin</code> All characters are hidden, except the last four are displayed. This option is equivalent to the <code>Social Insurance Number</code> option.</p>

Field Name	Field Type	Description
		<p>nino</p> <p>All characters are hidden. If the field contains nine characters, Salesforce automatically inserts spaces after each pair of characters. This option is equivalent to the <code>National Insurance Number</code> option.</p>
<code>picklist</code>	Picklist	If specified, the field is a picklist, and this field enumerates the picklist values and labels.
<code>populateExistingRows</code>	boolean	Indicates whether existing rows are populated (<code>true</code>) or not (<code>false</code>).
<code>precision</code>	int	The precision for number values. Precision is the number of digits in a number. For example, the precision value for the number 256.99 is 5.
<code>referenceTargetField</code>	string	Available only for indirect lookup relationship fields on external objects. Specifies the custom field on the parent object to match against this indirect lookup relationship field, whose values come from an external data source. The specified custom field on the parent object must have both <code>externalId</code> and <code>unique</code> set to <code>true</code> . This field is available in API version 32.0 and later.
<code>referenceTo</code>	string	If specified, indicates a reference this field has to another object.
<code>relationshipLabel</code>	string	Label for the relationship.
<code>relationshipName</code>	string	If specified, indicates the value for one-to-many relationships. For example, in the object <code>MyObject</code> that had a relationship to <code>YourObject</code> , the relationship name <code>YourObjects</code> makes the relationship type obvious.
<code>relationshipOrder</code>	int	<p>This field is valid for all master-detail relationships, but the value is only non-zero for junction objects. A junction object has two master-detail relationships, and is analogous to an association table in a many-to-many relationship. Junction objects must define one parent object as primary (0), the other as secondary (1). The definition of primary or secondary affects delete behavior and inheritance of look and feel, and record ownership for junction objects. For more information, see the Salesforce Help.</p> <p>0 or 1 are the only valid values, and 0 is always the value for objects that are not junction objects.</p>
<code>required</code>	boolean	Indicates whether the field requires a value on creation (<code>true</code>) or not (<code>false</code>).
<code>scale</code>	int	The scale for the field. Scale is the number of digits to the right of the decimal point in a number. For example, the number 256.99 has a scale of 2.
<code>startingNumber</code>	int	If specified, indicates the starting number for the field.
<code>stripMarkup</code>	boolean	Set to <code>true</code> to remove markup, or <code>false</code> to preserve markup. Used when converting a rich text area to a long text area.
<code>summarizedField</code>	string	Represents the field on the detail row that is being summarized. This field cannot be null unless the value of <code>SummaryOperation</code> is <code>count</code> .
<code>summaryForeignKey</code>	string	Represents the master-detail field on the child that defines the relationship between the parent and the child.

Field Name	Field Type	Description
summaryOperation	SummaryOperations is an enumeration of strings	Represents the sum operation to be performed. Valid values are: <ul style="list-style-type: none"> Count Min Max Sum
trackFeedHistory	boolean	Indicates whether the field is enabled for feed tracking (<code>true</code>) or not (<code>false</code>). This field is available in API version 18.0 and later.
trackHistory	boolean	Indicates whether history tracking is enabled for the field (<code>true</code>) or not (<code>false</code>). Also available for standard object fields (picklist and lookup fields only) in API version 30.0 and later.
trackTrending	boolean	Indicates whether historical trending data is captured for the field (<code>true</code>) or not (<code>false</code>). If this attribute is <code>true</code> for at least one field, the object is enabled for historical trending. Available in API version 29.0 and later.
trueValueIndexed	boolean	This field is only relevant for a checkbox field. If set, true values are built into the index. This field has been deprecated as of API version 14.0 and is only provided for backward compatibility.
type	FieldType is an enumeration of strings	Indicates the field type for the field. Valid values are: <ul style="list-style-type: none"> AutoNumber Lookup MasterDetail Checkbox Currency Date DateTime Email EncryptedText Number¹ Percent Phone Picklist MultiselectPicklist Summary Text TextArea LongTextArea Summary Url

Field Name	Field Type	Description
		<ul style="list-style-type: none"> • Hierarchy • File • CustomDataType • Html • Geolocation <p>For standard fields on standard objects, the <code>type</code> field is optional. This field is included for some standard field types, such as Picklist or Lookup, but not for others. The <code>type</code> field is included for custom fields.</p>
unique	boolean	Indicates whether the field is unique (<code>true</code>) or not (<code>false</code>).
visibleLines	int	Indicates the number of lines displayed for the field.
writeRequiresMasterRead	boolean	<p>Sets the minimum sharing access level required on the master record to create, edit, or delete child records. This field applies only to master-detail or junction object custom field types.</p> <ul style="list-style-type: none"> • <code>true</code>—Allows users with “Read” access to the master record permission to create, edit, or delete child records. This setting makes sharing less restrictive. • <code>false</code>—Allows users with “Read/Write” access to the master record permission to create, edit, or delete child records. This setting is more restrictive than <code>true</code>, and is the default value. <p>For junction objects, the most restrictive access from the two parents is enforced. For example, you set to <code>true</code> on both master-detail fields, but users have “Read” access to one master record and “Read/Write” access to the other master record. In this example, users are unable to create, edit, or delete child records.</p>

Picklist Metadata

Field Name	Field Type	Description
controllingField	string	The <code>fullName</code> of the controlling field if this field is a dependent picklist. A dependent picklist works with a controlling picklist or checkbox to filter the available options. The value chosen in the controlling field affects the values available in the dependent field. This field is available in API version 14.0 and later.
picklistValues	PicklistValue[]	Required. Represents a set of values for a picklist.
sorted	boolean	Required. Indicates whether values are sorted (<code>true</code>), or not (<code>false</code>).


PicklistValue

This metadata type defines a value in the picklist and specifies whether this value is the default value. This type extends `Metadata` and inherits its `fullName` field.

Note the following when working with picklist values:

- When you retrieve a standard object, you all picklist values are retrieved, not just the customized picklist values.
- When you deploy changes to standard picklist fields, picklist values are added as needed.
- You can't set a picklist value as inactive, but if the picklist value is missing and you invoke an `update()` call, the missing value becomes inactive.

Field Name	Field Type	Description
<code>allowEmail</code>	boolean	Indicates whether this value lets users email a quote PDF (<code>true</code>), or not (<code>false</code>). This field is only relevant for the <code>Status</code> field in quotes. This field is available in API version 18.0 and later.
<code>closed</code>	boolean	Indicates whether this value is associated with a closed status (<code>true</code>), or not (<code>false</code>). This field is only relevant for the standard <code>Status</code> field in cases and tasks. This field is available in API version 16.0 and later.
<code>color</code>	string	Indicates the color assigned to the picklist value when used in charts on reports and dashboards. The color is in hexadecimal format; for example, <code>#FF6600</code> . If a color is not specified, it's assigned dynamically during chart generation. This field is available in API version 17.0 and later.
<code>controllingFieldValues</code>	string[]	<p>A list of values in the controlling field that are linked to this picklist value. The controlling field can be a checkbox or a picklist. This field is available in API version 14.0 and later. The values in the list depend on the field type:</p> <ul style="list-style-type: none"> • Checkbox: <code>checked</code> or <code>unchecked</code>. • Picklist: The <code>fullName</code> of the picklist value in the controlling field.
<code>converted</code>	boolean	Indicates whether this value is associated with a converted status (<code>true</code>), or not (<code>false</code>). This field is relevant for only the standard <code>Lead Status</code> field in leads. Your organization can set its own guidelines for determining when a lead is qualified. The best practice is to convert a lead when it becomes a real opportunity that you want to forecast. For more information, see "Convert Qualified Leads" in the Salesforce online help. This field is available in API version 16.0 and later.
<code>cssExposed</code>	boolean	<p>Indicates whether this value is available in your Self-Service Portal (<code>true</code>), or not (<code>false</code>). This field is only relevant for the standard <code>Case Reason</code> field in cases.</p> <p>Self-Service provides an online support channel for your customers - allowing them to resolve their inquiries without contacting a customer service representative. For more information about Self-Service, see "Setting Up Self-Service" in the Salesforce online help.</p>

Field Name	Field Type	Description
		 Note: Starting with Spring '12, the Self-Service portal isn't available for new organizations. Existing organizations continue to have access to the Self-Service portal. This field is available in API version 16.0 and later.
default	boolean	Required. Indicates whether this value is the default picklist value in the specified picklist (<code>true</code>), or not (<code>false</code>).
description	string	Description of a custom picklist value. This field is only relevant for the standard <code>stage</code> field in opportunities. It is useful to include a description for a customized picklist value so that the historical reason for creating it can be tracked. This field is available in API version 16.0 and later.
forecastCategory	ForecastCategories, an enumeration of type string	Indicates whether this value is associated with a forecast category (<code>true</code>), or not (<code>false</code>). This field is only relevant for the standard <code>stage</code> field in opportunities. Valid values: <ul style="list-style-type: none"> • Omitted • Pipeline • BestCase • Forecast • Closed This field is available in API version 16.0 and later.
fullName	string	The name used as a unique identifier for API access. The <code>fullName</code> can contain only underscores and alphanumeric characters. It must be unique, begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. This field is inherited from Metadata.
highPriority	boolean	Indicates whether this value is a high priority item (<code>true</code>), or not (<code>false</code>). This field is only relevant for the standard <code>priority</code> field in tasks. For more information about tasks, see "Considerations in Using Tasks" in the Salesforce online help. This field is available in API version 16.0 and later.
probability	int	Indicates whether this value is a probability percentage (<code>true</code>), or not (<code>false</code>). This field is only relevant for the standard <code>stage</code> field in opportunities. For more information about opportunities, see "Opportunities Overview" in the Salesforce online help. This field is available in API version 16.0 and later.
reverseRole	string	A picklist value corresponding to a reverse role name for a partner. For example, for the role "subcontractor", a reverse role is "general contractor". Assigning a partner role to an account in Salesforce creates a reverse partner relationship so that both accounts list the other as a partner. This field is only relevant for partner roles.

Field Name	Field Type	Description
		For more information, see “Partner Fields” in the Salesforce online help. This field is available in API version 18.0 and later.
reviewed	boolean	Indicates whether this value is associated with a reviewed status (<code>true</code>), or not (<code>false</code>). This field is only relevant for the standard <code>Status</code> field in solutions. For more information about opportunities, see “Creating Solutions” in the Salesforce online help. This field is available in API version 16.0 and later.
won	boolean	Indicates whether this value is associated with a closed or won status (<code>true</code>), or not (<code>false</code>). This field is only relevant for the standard <code>Stage</code> field in opportunities. This field is available in API version 16.0 and later.

WorkflowFieldUpdate Metadata

For more information about WorkflowFieldUpdate, see the *Metadata API Developer's Guide*.

SOQL Limitations

This object doesn't support some SOQL operations.

COUNT()

Example Query: `SELECT COUNT() FROM EntityDefinition`

Error Returned: The "DurableId" field is of type number, but the value from the external data source is "Account.SearchResult". Contact your administrator for help.

GROUP BY

Example Query: `SELECT COUNT(qualifiedapiname), isfeedenabled FROM EntityDefinition GROUP BY isfeedenabled`

Error Returned: The requested operation is not yet supported by this SObject storage type, contact salesforce.com support for more information.

LIMIT, LIMIT OFFSET

Example Queries:

`SELECT qualifiedapiname FROM EntityDefinition LIMIT 5`

`SELECT qualifiedapiname FROM EntityDefinition LIMIT 5 OFFSET 10`

An incorrect result is returned because LIMIT and LIMIT OFFSET are ignored.

NOT

Example Query: `SELECT qualifiedapiname FROM EntityDefinition WHERE qualifiedapiname != 'Account'`

Error Returned: Only equals comparisons permitted

OR

Example Query: `SELECT qualifiedapiname, keyprefix FROM EntityDefinition WHERE isdeletable=true OR (isfeedenabled=false AND keyprefix='01j')`

Error Returned: Disjunctions not supported

FieldSet

Represents the metadata for a group of fields. Available from API version 33.0 or later.

Supported SOAP Calls

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

Supported REST HTTP Methods

GET, HEAD

Fields

Field	Details
Description	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The field set description. This can be useful to describe the reason for creating the set or its intended use.</p>
DeveloperName	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The API name of the field set.</p>
MasterLabel	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The set's label.</p>
NamespacePrefix	<p>Type string</p>

Field	Details
	Properties Filter, Group, Nillable, Sort
	Description The namespace of the package of which the field set is a part.

FlexiPage

Represents a Lightning Page. A Lightning Page is the home page for an app that appears as a menu item in the Salesforce1 navigation menu. Includes access to the associated FlexiPage object in the Salesforce Metadata API. Available from API version 31.0 or later.



Note: These app pages are known as FlexiPages in the API, but are referred to as Lightning Pages in the rest of the Salesforce documentation and UI.

Supported SOAP Calls

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

Supported REST HTTP Methods

GET, HEAD

Fields

Field	Details
Description	Type string
	Properties Filter, Group, Nillable, Sort
	Description The email page description. This can be useful to describe the reason for creating the page or its intended use.
DeveloperName	Type string
	Properties Filter, Group, Sort
	Description The API name of the Lightning Page.

Field	Details
FullName	<p>Type string</p> <p>Properties Create, Group, Nillable</p> <p>Description The full name of the associated FlexiPage object in Metadata API. Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
MasterLabel	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The page's label.</p>
Metadata	<p>Type FlexiPageMetadata</p> <p>Properties Create, Nillable, Update</p> <p>Description Lightning Page metadata. Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace of the package of which the flexipage is a part.</p>
SubjectType	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description This field is reserved for future use. Once the value of this field is set, it can't be changed.</p>

Field	Details
	Available in API version 33.0 or later.
Type	Type picklist Properties Filter, Group, Restricted picklistSort Description Required. The type of the Lightning Page Available in API version 32.0 or later. In API version 32.0, this field can only have a value of AppPage.

Sample Code

This code sample creates a new Lightning Page with a single Recent Items component, that shows recently used Accounts and MyCustomObject__cs

```

ComponentInstance recentItems = new ComponentInstance();
recentItems.setComponentName("flexipage:recentItems");
ComponentInstanceProperty cip = new ComponentInstanceProperty();
cip.setName("entityNames");
cip.setValue("Account,MyCustomObject__c");
recentItems.setComponentInstanceProperties(new ComponentInstanceProperty[]{cip});

FlexiPageRegion mainRegion = createRegion("main");
mainRegion.setComponentInstances(new ComponentInstance[] { recentItems });

FlexiPageMetadata fpMetadata = new FlexiPageMetadata();
fpMetadata.setFlexiPageRegions(new FlexiPageRegion[] {mainRegion});
fpMetadata.setMasterLabel("My FlexiPage");
fpMetadata.setDescription("A FlexiPage with a recent items component");

FlexiPage flexiPage = new FlexiPage();
flexiPage.setFullName("MyFlexiPageDevName");
flexiPage.setMetadata(fp);

// Create
SaveResult saveResult = soapConnection.create(new SObject[] { flexiPage });

```


Flow

Use the Flow object to retrieve and update specific flow versions.

With Flow, you can create an application that navigates users through a series of screens to query and update records in the database. You can also execute logic and provide branching capability based on user input to build dynamic applications. For information about the corresponding UI-based flow building tool, see [Cloud Flow Designer Overview](#) in the Salesforce Help.

When using the Tooling API to work with flows, consider that:

- You can describe information for a flow installed from a managed package but not its metadata.

- You can't overwrite an active flow or one that was once active.
 - Every time you update a flow, you're actually deleting the existing flow and creating a new flow from it, with a new ID.
-  **Note:** Legacy flows created with the Desktop Flow Designer can be queried and deleted, but they can't be updated, activated, or deactivated. Update your flow by recreating it with the Cloud Flow Designer.

You can delete a flow version, as long as it:

- Isn't active
- Doesn't have any paused or waiting interviews

If the flow version has paused or waiting interviews, wait for those interviews to finish, or delete them.

This object is available in API version 34.0 and later.

Supported SOAP Calls

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


Supported REST API HTTP Methods

DELETE, GET, HEAD, PATCH, POST

Fields

Field	Details
Definition	<p>Type FlowDefinition</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description This flow's definition object.</p>
DefinitionId	<p>Type ID</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of this flow's FlowDefinition.</p>
Description	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description A description of the flow, such as what it's meant to do or how it works.</p>

Field	Details
FullName	<p>Type string</p> <p>Properties Create, Group, Nillable</p> <p>Description The full name of the flow in the Metadata API. The <code>fullName</code> consists of two parts, separated by a hyphen:</p> <ul style="list-style-type: none"> • Unique name for the flow that contains only underscores and alphanumeric characters. It must be unique across the organization, begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. • Version number for the flow. <p>For example, "sampleFlow-3" specifies version 3 of the flow whose unique name is sampleFlow.</p> <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
MasterLabel	<p>Type string</p> <p>Properties Filter, Group, idLookup, Sort</p> <p>Description Label for the flow.</p>
Metadata	<p>Type mns : Flow</p> <p>Properties Create, Nillable, Update</p> <p>Description The flow's metadata. Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
ProcessType	<p>Type Restricted picklist</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The type of the flow:</p> <ul style="list-style-type: none"> • <code>AutoLaunchedFlow</code> • <code>Flow</code>


Field	Details
	<ul style="list-style-type: none"> Workflow LoginFlow ActionPlan JourneyBuilderIntegration UserProvisioningFlow <p> Note: This value has significant impact on validation when saving the flow and on the flow's runtime behavior. Don't change this value unless you understand the flow properties of the specified type.</p>
Status	<p>Type Restricted picklist</p> <p>Properties Filter, Group</p> <p>Description The flow's status:</p> <ul style="list-style-type: none"> Active Draft Obsolete InvalidDraft
VersionNumber	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The flow's version number.</p>

FlowDefinition

The parent of a set of flow versions.

When using the Tooling API to work with flow definitions, consider that:

- You can activate and deactivate flows with the `Metadata` field.
- You can update `masterlabel` and `description` of a `FlowDefinition`.
- `FlowDefinition` are implicitly created when the `Flow` object is created. This means `FlowDefinition` objects can only be updated.

 **Note:** Legacy flows created with the Desktop Flow Designer can be queried, activated, and deactivated, but they can't be updated or deleted. Update your flow by recreating it with the Cloud Flow Designer.

This object is available in API version 34.0 and later.

Supported SOAP API Calls

`query()`, `retrieve()`, `update()`

Supported REST API HTTP Methods

GET, HEAD, PATCH, POST

Fields

Field Name	Details
ActiveVersion	Type Flow Properties Filter, Group, Nillable, Sort Description The active flow version object.
ActiveVersionId	Type ID Properties Filter, Group, Nillable, Sort Description The ID of the active flow version.
Description	Type string Properties Nillable Description Flow definition information, specified by the organization's administrator.
DeveloperName	Type string Properties Filter, Group, Sort Description Developer name of this flow definition.
FullName	Type string Properties Create, Group, Nillable

Field Name	Details
	<p>Description</p> <p>The full name of the flow definition in the Metadata API.</p> <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
LatestVersion	<p>Type</p> <p>Flow</p> <p>Properties</p> <p>Filter, Group, Nillable, Sort</p> <p>Description</p> <p>The latest flow version object, regardless of the flow's state.</p>
LatestVersionId	<p>Type</p> <p>ID</p> <p>Properties</p> <p>Filter, Group, Nillable, Sort</p> <p>Description</p> <p>ID of the latest flow version, regardless of the flow's state.</p>
MasterLabel	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Nillable, Sort</p> <p>Description</p> <p>Label for the flow definition.</p>
Metadata	<p>Type</p> <p>mns : FlowDefinition</p> <p>Properties</p> <p>Create, Nillable, Update</p> <p>Description</p> <p>The flow definition's metadata object, containing information about which flow version is active and the flow definition's description.</p> <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
NamespacePrefix	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Nillable, Sort</p>

Field Name	Details
	Description The namespace associated with this flow definition.

HeapDump

A complex type that represents a heap dump in an ApexExecutionOverlayResult object. Available from API version 28.0 or later.

Fields

Field	Details
className	Type string Description The name of the Apex class or trigger.
extents	Type array of TypeExtent Description TypeExtent includes the following fields: <ul style="list-style-type: none">• collectionType• count• definition (array of AttributeDefinition)• extent (array of HeapAddress)• totalSize• typeName
heapDumpDate	Type dateTime Description The date and time that the heap dump was captured.
namespace	Type string Description The namespace of the Apex class or trigger. Null if there is no namespace.

Usage

Use heap dumps to capture structured debugging information.

HistoryRetentionJob

Represents the body of retained data from the archive, and the status of the archived data. Available in API version 29.0 or later.

Supported SOAP API Calls

`describeSObjects()`, `query()`

Supported REST API HTTP Methods

GET

Fields

Field Name	Details
DurationSeconds	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description How many seconds the field history retention job took to complete (whether successful or not).</p>
HistoryType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The object type that contains the field history that you retained. Valid values for standard objects are:</p> <ul style="list-style-type: none">• Account• Case• Contact• Leads• Opportunity <p>For custom objects, use the object name.</p>

Field Name	Details
NumberOfRowsRetained	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of field history rows that a field history retention job has retained.</p>
RetainOlderThanDate	<p>Type dateTime</p> <p>Properties Filter, Sort</p> <p>Description The date and time before which all field history data was retained.</p>
StartDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The start date of the field history retention job.</p>
Status	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Provides the status of the field history retention job. By default, the pilot feature copies data to the archive, leaving a duplicate of the archived data in Salesforce. Deletion of data from Salesforce after archiving is available upon request. Status can include:</p> <ul style="list-style-type: none">• CopyScheduled• CopyRunning• CopySucceeded• CopyFailed• CopyKilled• NothingToArchive• DeleteScheduled• DeleteRunning• DeleteSucceeded• DeleteFailed

Field Name	Details
	<ul style="list-style-type: none"> DeleteKilled

Layout

Represents a page layout. For more information, see “Manage Page Layouts” in the Salesforce online help.

This object is available in API version 32.0 and later.

Supported SOAP Calls

`query()`, `retrieve()`

Supported REST HTTP Methods

GET

Fields


Field	Details
EntityDefinitionId	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The Id of the EntityDefinition object associated with this object.</p>
FullName	<p>Type string</p> <p>Properties Create, Group, Nillable</p> <p>Description The unique name of the layout used as the identifier for API access. The <code>fullName</code> can contain only underscores and alphanumeric characters. It must be unique, begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores.</p> <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
Metadata	<p>Type LayoutMetadata</p>

Field	Details
	<p>Properties Create, Nillable, Update</p> <p>Description Layout metadata.</p> <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
Name	<p>Type string</p> <p>Properties Filter, Group, idLookup, Nillable, Sort</p> <p>Description The layout name.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description A unique string to distinguish this layout from any others. For example, if this layout is being using by a flow, use the <code>NamespacePrefix</code> to uniquely identify the layouts in multiple flow instances.</p>
ShowSubmitAndAttachButton	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Only allowed on Case layout. If true, Submit & Add Attachment displays on case edit pages to portal users in the Customer Portal.</p>
TableEnumOrId	<p>Type string</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The enum (for example, Account) or ID of the object this layout is on.</p>

LookupFilter

Represents a lookup filter, which restricts the valid values and lookup dialog results for lookup, master-detail, and hierarchical relationship fields.

Available from Tooling API version 34.0 or later.

 **Note:** LookupFilter is not supported on the article type object.

Supported SOAP Calls

`query()`

Supported REST HTTP Methods

GET

Fields

Field Name	Details
Active	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, the lookup filter is active.</p>
DeveloperName	<p>Type string</p> <p>Properties Filter, Group, Namefield, Sort</p> <p>Description Required. The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your organization. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Label is Record Type Name.</p>
FullName	<p>Type string</p> <p>Properties Filter, Group, Sort</p>

Field Name	Details
	<p>Description</p> <p>The full name of the associated metadata object in Metadata API.</p> <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
IsOptional	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Defaulted on create, Filter, Group, Sort</p> <p>Description</p> <p>Required. If <code>true</code>, the lookup filter is optional.</p>
Metadata	<p>Type</p> <p>LookupFilter</p> <p>Properties</p> <p>Create, Nillable, Update</p> <p>Description</p> <p>The metadata for this lookup filter.</p> <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
NamespacePrefix	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Nillable, Sort</p> <p>Description</p> <p>The namespace of the custom field, which might be different from the object's namespace.</p>
SourceFieldDefinition	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>The field whose value is restricted by the filter.</p>
SourceFieldDefinitionId	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Sort</p>

Field Name	Details
	Description Durable ID of the object specified in <code>SourceFieldDefinition</code> .
<code>SourceObject</code>	Type string Properties Filter, Group, Sort Description The object that contains the lookup field that uses this lookup filter. Null if the lookup filter doesn't reference fields on the source object.
<code>TargetEntityDefinition</code>	Type EntityDefinition Properties Filter, Group, Sort Description The entity definition for the source lookup field.
<code>TargetEntityDefinitionId</code>	Type string Properties Filter, Group, Sort Description ID of the <code>TargetEntityDefinition</code> .



Example: The Owner field on Account points to a user with certain characteristics.

In this case, `SourceFieldDefinition` is `Account.Owner` and `TargetEntityDefinition` is `User`, because `Owner` is a lookup field to `User`.

LookupFilter Metadata

Metadata about the lookup filter is returned in the `Metadata` field:

Field	Type	Description
<code>active</code>	boolean	Required. If <code>true</code> , the lookup filter is active.
<code>booleanFilter</code>	string	The filter logic, if any, applied to this filter using Boolean operators AND, OR, or NOT.
<code>description</code>	string	A description of the filter does.
<code>errorMessage</code>	string	The error message that appears if the lookup filter fails.

Field	Type	Description
filterItems	FilterItem	Required. The set of filter conditions. Each lookup filter can have up to 10 FilterItems.
infoMessage	string	Information displayed on the page to help the user. For example, explaining why some items are excluded in the lookup filter.
isOptional	boolean	Required. If <code>true</code> , the lookup filter is optional.

FilterItem Metadata

Represents one entry in a set of filter criteria.

Field	Type	Description
field	string	Represents the field specified in the filter.
operation	FilterOperation, an enumeration of strings	Valid values: <ul style="list-style-type: none"> • <code>equals</code> • <code>notEqual</code> • <code>lessThan</code> • <code>greaterThan</code> • <code>lessOrEqual</code> • <code>greaterOrEqual</code> • <code>contains</code> • <code>notContain</code> • <code>startsWith</code> • <code>includes</code> • <code>excludes</code> • <code>within</code> (DISTANCE criteria only)
value	string	Represents the value of the filter item being operated upon. For example, if the filter is <code>my_number_field__c > 1</code> , the value of this field is <code>1</code> .
valueField	string	Specifies if the final column in the filter contains a field or a field value. Approval processes don't support this field in filter criteria.

MenuItem

Represents a menu item.

This object is available in API version 32.0 and later.

Supported SOAP Calls

`query()`, `update()`

Supported REST HTTP Methods

GET, POST

Fields

Field	Details
Active	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the item in the menu is active (<code>true</code>) or not (<code>false</code>).</p>
AppId	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the app that this menu item is associated with. Can be an enum (such as Feed or People) or an alphanumeric ID. Use <code>AppId</code> as the unique ID for the menu item, not <code>Id</code>.</p>
Color	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The color of the menu item that appears in the user interface. This field is described in Web color RGB format, such as <code>00FF00</code>.</p>
IconURL	<p>Type url</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The URL of an icon in the menu item.</p>
Label	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field	Details
	Description The menu item label that appears in the user interface.
MenuType	Type picklist Properties Filter, Group, Nillable, Restricted picklist, Sort Description The type of menu that this menu item belongs to. Valid values are: <ul style="list-style-type: none"> • <code>AppSwitcher</code>: the Force.com app menu, a drop-down menu that's displayed at the top of every app page • <code>Salesforce1</code>: the Salesforce1 navigation menu • <code>NetworkTabs</code>: the Salesforce Communities tab set This field is required for <code>query()</code> .
SortOrder	Type int Properties Filter, Group, Nillable, Sort, Update Description The <code>SortOrder</code> value determines the order in which a menu item is displayed in the user interface. This field must be an ordinal number greater than 0, and must be unique in the list. Inactive menu items have a value of -1.
Theme	Type string Properties Filter, Group, Nillable, Sort Description The associated theme, which must be one of the following values. <ul style="list-style-type: none"> • <code>theme2</code>: the Salesforce theme that was used prior to Spring '10 • <code>theme3</code>: the Salesforce theme that was introduced in Spring '10 • <code>theme4</code>: the theme that was introduced in Winter '14 for the mobile touchscreen version of Salesforce • <code>custom</code>: the theme that's associated with a custom icon This field is required for <code>query()</code> for <code>Color</code> and <code>IconURL</code> .

Usage

MenuItem can be queried and manipulated to change how menu items appear in Salesforce. The following example modifies the Salesforce1 left navigation menu.

```
String query = "SELECT AppId, Label, Active, SortOrder FROM MenuItem "
+
    "WHERE MenuType = 'Salesforce1'";
SObject[] records = sforce.query(query).getRecords();

//Activate all menu items
for (int i = 0; i < records.length; i++) {
    MenuItem item = (MenuItem)records[i];
    item.setOrder(i + 1);
    item.setActive(true);
}

sforce.update(records);
```

MetadataContainer

Manages working copies of ApexClassMember, ApexTriggerMember, ApexPageMember, and ApexComponentMember objects, including collections of objects to be deployed together.

Supported SOAP API Calls

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

Supported REST API HTTP Methods

Query, GET, POST, PATCH, DELETE

Fields

Field Name	Details
Name	<p>Type</p> <p>string</p> <p>Properties</p> <p>Create, Filter, Group, Sort, Update</p> <p>Description</p> <p>The name of the MetadataContainer. If a container with the same name already exists, an error is reported on <code>create()</code> or POST.</p> <p>This field is required.</p>

Usage

Use a `MetadataContainer` as a package for your tool's workspace. As a user works in the tool, update the [ApexClassMember](#), [ApexTriggerMember](#), [ApexPageMember](#) and [ApexComponentMember](#) objects in the `MetadataContainer` and use a [ContainerAsyncRequest](#) object to save and deploy changes to the current organization.

A `MetadataContainer` can be reused, but container members can't.

- When a `ContainerAsyncRequest` completes successfully, the `MetadataContainerId` field on all container members is changed from the ID of the `MetadataContainer` to the ID of the `ContainerAsyncRequest`. At this point, container members can no longer be modified or deployed, and can't be queried via the `MetadataContainer`; you have to query the `ContainerAsyncRequest` to see what was deployed.
- If the deployment fails, container members remain on the `MetadataContainer` and can still be modified until they are successfully deployed on another `ContainerAsyncRequest`. The `MetadataContainerId` field on the completed (failed deployment) `ContainerAsyncRequest` is set to the ID of the `MetadataContainer`, so you can have multiple completed `ContainerAsyncRequests` on a single `MetadataContainer`.



Note: Deleting a `MetadataContainer` deletes all objects that reference it.

Profile

Represents a user profile. A profile defines a user's permission to perform different functions within Salesforce.

This object is available in API version 32.0 and later.

Supported SOAP Calls

`getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`

Supported REST HTTP Methods

GET, PATCH

Fields

Field	Details
Description	Type string Properties Filter, Group, Nillable, Sort Description The profile description, limited to 255 characters.
FullName	Type string

Field	Details
	<p>Properties Create, Group, Nillable</p> <p>Description The unique profile name. Use this name when creating the profile, before you have an ID. This name can contain only underscores and alphanumeric characters, and must be unique in your organization. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores.</p> <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
Metadata	<p>Type ProfileMetadata</p> <p>Properties Create, Nillable, Update</p> <p>Description The profile metadata.</p> <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
Name	<p>Type string</p> <p>Properties Filter, Group, idLookup, Sort</p> <p>Description The profile name.</p>

ProfileLayout

Represents a profile layout.

This object is available in API version 32.0 and later.

Supported SOAP Calls

`query()`, `retrieve()`

Supported REST HTTP Methods

GET

Fields

Field	Details
LayoutId	Type ID Properties Filter, Group, Sort Description The unique identifier for this layout.
ProfileId	Type ID Properties Filter, Group, Sort Description The unique identifier for this profile.
RecordTypeId	Type ID Properties Filter, Group, Sort Description The unique identifier for the record.
TableEnumOrId	Type string Properties Filter, Group, Restricted picklist, Sort Description The enum (for example, Account) or ID of the object this field is on.

Publisher

Represents the publisher of objects and fields. For example, Salesforce is the publisher for standard objects, the organization is the publisher for custom objects, and the package is the publisher for installed packages. Available in Tooling API version 34.0 and later.

Fields

Field	Details
DurableId	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Unique identifier for the field. Always retrieve this value before using it, as the value isn't guaranteed to stay the same from one release to the next. Use this field to simplify queries.</p>
InstalledEntityDefinitions	<p>Type QueryResult</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Metadata for the objects installed by this publisher.</p>
InstalledFieldDefinitions	<p>Type QueryResult</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Metadata for the fields installed by this publisher.</p>
IsSalesforce	<p>Type boolean</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Indicates whether Salesforce provided the associated objects or fields (<code>true</code>).</p>
Name	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The presentation-friendly name of the publisher.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field	Details
	<p>Description</p> <p>The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the <code>namespacePrefix__componentName</code> notation.</p> <p>The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. In organizations that are not Developer Edition organizations, <code>NamespacePrefix</code> is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects. <p><code>NamespacePrefix</code> is null if the publisher is Salesforce.</p>

SOQL Limitations

This object doesn't support some SOQL operations.

COUNT()

Example Query: `SELECT COUNT() FROM EntityDefinition`

Error Returned: The "DurableId" field is of type number, but the value from the external data source is "Account.SearchResult". Contact your administrator for help.

GROUP BY

Example Query: `SELECT COUNT(qualifiedapiname), isfeedenabled FROM EntityDefinition GROUP BY isfeedenabled`

Error Returned: The requested operation is not yet supported by this SObject storage type, contact salesforce.com support for more information.

LIMIT, LIMIT OFFSET

Example Queries:

`SELECT qualifiedapiname FROM EntityDefinition LIMIT 5`

`SELECT qualifiedapiname FROM EntityDefinition LIMIT 5 OFFSET 10`

An incorrect result is returned because LIMIT and LIMIT OFFSET are ignored.

NOT

Example Query: `SELECT qualifiedapiname FROM EntityDefinition WHERE qualifiedapiname!='Account'`

Error Returned: Only equals comparisons permitted

OR

Example Query: `SELECT qualifiedapiname, keyprefix FROM EntityDefinition WHERE isdeletable=true OR (isfeedenabled=false AND keyprefix='01j')`

Error Returned: Disjunctions not supported

QueryResult

Represents the results of a query. For example, if you query on the object `EntityDefinition`, all the layouts for that entity are returned as an array of `QueryResult` objects in the `Layouts` field. Available in Tooling API version 34.0 and later.

`QueryResult` is not an extension of `sObject`.

Fields

Field	Details
<code>done</code>	<p>Type boolean</p> <p>Description If <code>true</code>, no additional rows can be retrieved from the query result. If <code>false</code>, one or more rows remain to be retrieved. Use this value as a loop condition while iterating through query results.</p>
<code>entityTypeName</code>	<p>Type string</p> <p>Description The object or entity type, such as <code>ApexClass</code> or <code>CompactLayoutInfo</code>.</p>
<code>nextRecordsUrl</code>	<p>Type string</p> <p>Description If the results exceed the current batch size, this field contains the URL of the next record in the query result set. This field is populated for the REST resource <code>queryAll</code>, and is analogous to <code>queryLocator</code> for SOAP calls.</p>
<code>queryLocator</code>	<p>Type <code>QueryLocator</code></p> <p>Description If the results exceed the current batch size, this field contains a unique identifier used to retrieve the next batch of records. This field is populated for SOAP <code>queryMore()</code> and is analogous to the REST resource <code>queryAll</code>. Each new batch returns a new <code>queryLocator</code> value.</p>
<code>records</code>	<p>Type <code>sObject</code></p> <p>Description Array of <code>sObjects</code> matching the data specified in the query.</p>

Field	Details
<code>size</code>	<p>Type int</p> <p>Description Total number of rows returned. If no rows were returned, the value is (0). This field is the same as the <code>size</code> field in <code>QueryResult</code> in the Enterprise and Partner WSDLs.</p>
<code>totalSize</code>	<p>Type int</p> <p>Description Total number of rows returned. Indicates whether the query retrieved any rows (any value greater than 0) or not (0). This field is the same as the <code>totalSize</code> field in <code>QueryResult</code> using the REST resource <code>query</code> or <code>queryAll</code>.</p>

QueryLocator Metadata

Field	Details
<code>queryLocator</code>	<p>Type string</p> <p>Description If the results exceed the current batch size, this field contains an identifier. Use with the SOAP <code>queryMore()</code> call to retrieve the next batch of records. Each new batch returns a new <code>queryLocator</code> value.</p>

QuickActionDefinition

Represents the definition of a quick action.

This object is available in API version 32.0 and later.

Supported SOAP Calls

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

Supported REST HTTP Methods

DELETE, GET, PATCH, POST

Fields

Field	Details
Description	<p>Type textarea</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The description of the action.</p>
DeveloperName	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The unique name of the action in the API. This field corresponds to the Name field in the user interface.</p>
Height	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The height of the action, in pixels. This field is set only when the quick action has a custom icon.</p>
IconId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ID of the action icon. This field is set only when the quick action has a custom icon.</p>
Label	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The action label that corresponds to the Label field in the user interface.</p>
Language	<p>Type picklist</p>

Field	Details
	<p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The language of the action. Valid values are:</p> <ul style="list-style-type: none"> Chinese (Simplified): zh_CN Chinese (Traditional): zh_TW Danish: da Dutch: nl_NL English: en_US Finnish: fi French: fr German: de Italian: it Japanese: ja Korean: ko Norwegian: no Portuguese (Brazil): pt_BR Russian: ru Spanish: es Spanish (Mexico): es_MX Swedish: sv Thai: th
MasterLabel	<p>Type string</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description The action label.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace of the action.</p>
SubjectType	<p>Type picklist</p>

Field	Details
	<p>Properties Create, Filter, Group, Restricted picklist, Sort</p> <p>Description The associated object's API name. For example, <code>FeedItem</code>.</p>
StandardLabel	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The standard label for the action. Valid values are:</p> <ul style="list-style-type: none"> • <code>ChangeDueDate</code> • <code>ChangePriority</code> • <code>ChangeStatus</code> • <code>CreateNew</code> • <code>CreateNewRecordType</code> • <code>Defer</code> • <code>EditDescription</code> • <code>LogACall</code> • <code>LogANote</code> • <code>New</code> • <code>NewChild</code> • <code>NewChildRecordType</code> • <code>NewRecordType</code> • <code>Quick</code> • <code>QuickRecordType</code> • <code>SendEmail</code> • <code>SocialPost</code> • <code>Update</code>
TargetField	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The API name of the parent object for the record created by this quick action. For example, <code>CollaborationGroup</code>.</p>
TargetRecordTypeId	<p>Type reference</p>

Field	Details
	<p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ID of the target record type.</p>
TargetSubjectType	<p>Type picklist</p> <p>Properties Create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description The API name of the type of object record this action will create. For example, OpportunityLineItem.</p>
Type	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The type of action. Valid values are:</p> <ul style="list-style-type: none"> • Canvas • Create • LogACall • Post • SendEmail • SocialPost • Update • VisualforcePage
Width	<p>Type int</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The width of the action, in pixels. This field is set only when the quick action has a custom icon.</p>

Usage

A QuickActionDefinition represents information about a quick action. The following example creates a global quick action that lets users quickly create a task.

```
QuickActionDefinition qad = new QuickActionDefinition();
qad.setDeveloperName("MyQuickCreateTaskAction");
qad.setObjectType("Global");
qad.setTargetObjectType("Task");
qad.setMasterLabel("Quick create a task");
qad.setType(QuickActionType.Create);
qad.setDescription("Quickly creates a Task");

sforce.create(new SObject[] {qad});
```

QuickActionList

Represents a list of quick actions.

This object is available in API version 32.0 and later.

Supported SOAP Calls

create(), query(), retrieve(), update(), upsert()

Supported REST HTTP Methods

DELETE, GET, PATCH, POST

Fields

Field	Details
LayoutId	Type reference Properties Create, Filter, Group, Sort Description The ID of the associated layout.

Usage

A QuickActionList is a junction between QuickActionListItem objects and a layout. If a layout doesn't have an associated QuickActionList, it inherits the actions from the global page layout.

The following example retrieves all quick action lists in an organization and their associated layout ID.

```
String query = "SELECT Id,LayoutId FROM QuickActionList";
SObject[] records = sforce.query(query).getRecords();

for (int i = 0; i < records.length; i++) {
    QuickActionList list = (QuickActionList)records[i];
    String relatedLayoutId = list.get("LayoutId");
}
```

QuickActionListItem

Represents an item in a quick action list.

This object is available in API version 32.0 and later.

Supported SOAP Calls

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

Supported REST HTTP Methods

DELETE, GET, PATCH, POST

Fields

Field	Details
QuickActionDefinition	<p>Type</p> <p>picklist</p> <p>Properties</p> <p>Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description</p> <p>The enum name or ID of the QuickActionDefinition that's associated with this list item. Valid values are:</p> <ul style="list-style-type: none"> • <code>Case.ChangeStatus</code> • <code>Case.LogACall</code> • <code>FeedItem.ContentPost</code> • <code>FeedItem.LinkPost</code> • <code>FeedItem.MobileSmartActions</code> • <code>FeedItem.PollPost</code> • <code>FeedItem.QuestionPost</code> • <code>FeedItem.TextPost</code>

Field	Details
QuickActionListId	Type reference Properties Create, Filter, Group, Sort Description The ID of the QuickActionList associated with this list item.
SortOrder	Type int Properties Create, Filter, Group, Sort, Update Description The order in which this list item appears in the picklist. This field must be an ordinal number greater than 0, and must be unique in the list.

Usage

A QuickActionListItem associates a QuickActionDefinition with a QuickActionList. You can query to find out which quick actions are in a list, insert or delete to add or remove quick actions from a list, and update to change the order of quick actions in the list.

The following example reverses the order in the list of the actions, and then removes the first action from the list.

```
String query = "SELECT Id,SortOrder FROM QuickActionListItem Where QuickActionListId='" +
    listId + "'";
SObject[] records = sforce.query(query).getRecords();

for(int i=0;i<records.length;i++) {
    QuickActionListItem item = (QuickActionListItem)records[i];
    item.setSortOrder(records.length-i);
}

sforce.update(records);

// Last record in array is first record in reordered list
sforce.delete(records[records.length-1].getId());
```

RecentlyViewed

Represents metadata entities typically found in Setup such as page layout definitions, workflow rule definitions, and email templates that the current user has recently viewed.

This object is available in the Tooling API version 33.0 and later.

Supported SOAP Calls

query(), update()

Supported REST HTTP Methods

GET

Special Usage Rules

The RecentlyViewed object supports the following metadata entities:

- Apex classes
- Apex triggers
- Approval processes
- Apps
- Custom report types
- Email templates
- Fields
- Objects
- Page layouts
- Permission sets
- Profiles
- Static resources
- Tabs
- Users
- Validation rules
- Visualforce pages
- Visualforce components
- Workflow email alerts
- Workflow field updates
- Workflow outbound messages
- Workflow rules
- Workflow tasks

Fields

Field	Details
Alias	<div><div>Type</div><div>string</div><div>Properties</div><div>Filter, Group, Nillable, Sort</div><div>Description</div><div>The alias on the item.</div></div>

Field	Details
Email	<p>Type email</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The email address on the item.</p>
FirstName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The first name on the item.</p>
Id	<p>Type ID</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description The ID of the recently viewed item.</p>
IsActive	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the recently viewed item is an active user (true) or not (false). This field contains a value only if the recently viewed item is a user.</p>
LastName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The last name on the item.</p>
LastReferencedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort, Update</p>

Field	Details
	Description The timestamp for when the current user last viewed an item related to this item.
LastViewedDate	Type dateTime Properties Filter, Nillable, Sort, Update Description The timestamp for when the current user last viewed this item. If this value is null, this item might only have been referenced (see <code>LastReferencedDate</code>) and not viewed.
Name	Type string Properties Filter, Group, Nillable, Sort Description If the recently viewed item is a user, this is the user's name. Specifically, it's the concatenation of the <code>FirstName</code> and <code>LastName</code> field values.
NetworkId	Type reference Properties Filter, Group, Nillable, Sort Description The ID of the community that this group is part of. This field is available only if Salesforce Communities is enabled in your organization.
Phone	Type phone Properties Filter, Group, Nillable, Sort Description The phone number on the item.
ProfileId	Type reference Properties Filter, Group, Nillable, Sort Description If the recently viewed item is a user, this is the user's profile ID.

Field	Details
RelatedObject	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The object that the recently viewed item is related to. For example, if the recently viewed item is an Account Custom Field, then the related object will be Account. Not all recently viewed items will have a related object.</p>
Title	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description If the recently viewed item is a user, this is the user's title. For example, CFO or CEO.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The sObject type for this recently viewed item.</p>
UserRoleId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the user role associated with this object.</p>

Usage

This object provides a heterogeneous list of different metadata types and consists of recently viewed records. A record is considered viewed when the user sees the details associated with it, but not when the user sees it in a list with other records. Use this object to programmatically construct a list of recently viewed items specific to the current user, for example, on a custom user interface or for search auto-complete options. You can also retrieve a filtered list of records by object type (`Type`). The RecentlyViewed data is periodically truncated down to 200 records per object.

Use this query in your code to retrieve a list of all the records that were recently viewed. The results are ordered from most to least recent.

```
SELECT Id, Name
FROM RecentlyViewed
```



```
WHERE LastViewedDate !=null
ORDER BY LastViewedDate DESC
```

Use this query to retrieve data that was either viewed or referenced, but only for a limited set of objects.

```
SELECT Id, Name
FROM RecentlyViewed
WHERE Type IN ('CustomEntityDefinition', 'CustomFieldDefinition')
ORDER BY LastViewedDate DESC
```

RecordType

Represents a custom record type.

This object is available in API version 32.0 and later.

Supported SOAP Calls

`create()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `update()`, `upsert()`

Supported REST HTTP Methods

GET, PATCH, POST

Fields

Field	Details
BusinessProcessId	Type ID Properties Create, Filter, Group, Nillable, Sort, Update Description ID of an associated BusinessProcess.
Description	Type string Properties Filter, Group, Nillable, Sort Description The record type description, limited to 255 characters.
EntityDefinitionId	Type string Properties Filter, Group, Sort

Field	Details
	Description The ID of the entity containing the record.
FullName	Type string Properties Create, Group, Nillable Description The full name of the associated metadata object in Metadata API. Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.
IsActive	Type boolean Properties Defaulted on create, Filter, Group, Sort, Update Description Indicates whether this record is active (<code>true</code>) or not (<code>false</code>). Only active record types can be applied to records.
Metadata	Type RecordTypeMetadata Properties Create, Nillable, Update Description Record metadata. Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.
Name	Type string Properties Nillable Description The record type name.
NamespacePrefix	Type string Properties Nillable

Field	Details
	Description A unique string to distinguish this type from any others.
SubjectType	Type string Properties Filter, Group, Nillable, Sort Description The type of standard object that this record type is derived from.

RelationshipDomain

Represents the relationship an object has with other objects. Use this object in queries to answer the question “which objects are the child objects for the object defined in `ParentSubject`.” Available in Tooling API version 34.0 and later.

Fields

Field	Details
ChildSubject	Type EntityDefinition Properties Filter, Group, Sort Description Metadata for the child object, if any.
ChildSubjectId	Type string Properties Filter, Group, Nillable, Sort Description ID of the <code>ChildSubject</code> .
DurableId	Type string Properties Filter, Group, Nillable, Sort Description Unique identifier for the field. Always retrieve this value before using it, as the value isn’t guaranteed to stay the same from one release to the next. Use this field to simplify queries.

Field	Details
Field	<p>Type FieldDefinition</p> <p>Properties Filter, Group, Sort</p> <p>Description The relationship field on this object that defines the relationship to <code>ChildSubject</code> or <code>ParentSubject</code>.</p>
FieldId	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of <code>Field</code>.</p>
IsCascadeDelete	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, this object's parent can't be deleted until all records for this object are deleted. Corresponds to <code>Cascade</code> value for <code>DeleteConstraint</code> in the Metadata API.</p>
IsDeprecatedAndHidden	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, this object is unavailable for the current version.</p>
IsRestrictedDelete	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, this object can't be deleted. Corresponds to <code>Restrict</code> value for <code>DeleteConstraint</code> in the Metadata API.</p>
ParentSubject	<p>Type EntityDefinition</p> <p>Properties Filter, Group, Sort</p>

Field	Details
	Description Metadata for the parent object, if any.
ParentSubjectId	Type string Properties Filter, Group, Nillable, Sort Description ID of the ParentSubject.
RelationshipInfo	Type RelationshipInfo Properties Filter, Group, Sort Description Properties about the relationship.
RelationshipInfoId	Type string Properties Filter, Group, Nillable, Sort Description ID of RelationshipInfo for this relationship domain.
RelationshipName	Type string Properties Filter, Group, Nillable, Sort Description Name of this relationship.

SOQL Limitations

This object doesn't support some SOQL operations.

COUNT()

Example Query: `SELECT COUNT() FROM EntityDefinition`

Error Returned: The "DurableId" field is of type number, but the value from the external data source is "Account.SearchResult". Contact your administrator for help.

GROUP BY

Example Query: `SELECT COUNT(qualifiedapiname), isfeedenabled FROM EntityDefinition GROUP BY isfeedenabled`

Error Returned: The requested operation is not yet supported by this SObject storage type, contact salesforce.com support for more information.

LIMIT, LIMIT OFFSET

Example Queries:

```
SELECT qualifiedapiname FROM EntityDefinition LIMIT 5
```

```
SELECT qualifiedapiname FROM EntityDefinition LIMIT 5 OFFSET 10
```

An incorrect result is returned because LIMIT and LIMIT OFFSET are ignored.

NOT

Example Query: `SELECT qualifiedapiname FROM EntityDefinition WHERE qualifiedapiname != 'Account'`

Error Returned: Only equals comparisons permitted

OR

Example Query: `SELECT qualifiedapiname, keyprefix FROM EntityDefinition WHERE isdeletable=true OR (isfeedenabled=false AND keyprefix='01j')`

Error Returned: Disjunctions not supported

RelationshipInfo

Represents the properties of a relationship between objects. Use this object in queries to answer the question “which objects are parent objects for the object defined in `ChildSobject`. Available in Tooling API version 34.0 and later.

Fields

Field	Details
ChildSobject	Type EntityDefinition Properties Filter, Group, Sort Description Metadata for the child object, if any.
ChildSobjectId	Type string Properties Filter, Group, Nillable, Sort Description ID of the <code>ChildSobject</code> .
DurableId	Type string

Field	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description Unique identifier for the field. Always retrieve this value before using it, as the value isn't guaranteed to stay the same from one release to the next. Use this field to simplify queries.</p>
Field	<p>Type FieldDefinition</p> <p>Properties Filter, Group, Sort</p> <p>Description The relationship field that defines the relationship to <code>ChildSubject</code> or <code>ParentSubject</code>.</p>
FieldId	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of <code>Field</code>.</p>
IsCascadeDelete	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, this object's parent can't be deleted until all records for this object are deleted. Corresponds to <code>Cascade</code> value for <code>DeleteConstraint</code> in the Metadata API.</p>
IsDeprecatedAndHidden	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, this object is unavailable for the current version.</p>
IsRestrictedDelete	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p>

Field	Details
	Description If <code>true</code> , this object can't be deleted. Corresponds to <code>Restrict</code> value for <code>DeleteConstraint</code> in the Metadata API.
RelationshipDomains	Type QueryResult Properties Filter, Group, Sort Description The RelationshipDomain records associated with this object.

SOQL Limitations

This object doesn't support some SOQL operations.

COUNT()

Example Query: `SELECT COUNT() FROM EntityDefinition`

Error Returned: The "DurableId" field is of type number, but the value from the external data source is "Account.SearchResult". Contact your administrator for help.

GROUP BY

Example Query: `SELECT COUNT(qualifiedapiname), isfeedenabled FROM EntityDefinition GROUP BY isfeedenabled`

Error Returned: The requested operation is not yet supported by this SObject storage type, contact salesforce.com support for more information.

LIMIT, LIMIT OFFSET

Example Queries:

`SELECT qualifiedapiname FROM EntityDefinition LIMIT 5`

`SELECT qualifiedapiname FROM EntityDefinition LIMIT 5 OFFSET 10`

An incorrect result is returned because LIMIT and LIMIT OFFSET are ignored.

NOT

Example Query: `SELECT qualifiedapiname FROM EntityDefinition WHERE qualifiedapiname != 'Account'`

Error Returned: Only equals comparisons permitted

OR

Example Query: `SELECT qualifiedapiname, keyprefix FROM EntityDefinition WHERE isdeletable=true OR (isfeedenabled=false AND keyprefix='01j')`

Error Returned: Disjunctions not supported

SearchLayout

Represents a search layout defined for an object.

This object is available in the Tooling API version 34.0 and later.

Supported Calls

`query()`

Supported REST Methods

GET

Fields

Field	Details
ButtonsDisplayed	<p>Type SearchLayoutButtonsDisplayed</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The list of buttons available in list views for an object. This field is equivalent to the <code>Buttons Displayed</code> value in Object Name List View in the Search Layouts related list on the object detail page. It's also equivalent to the <code>ListViewButtons</code> field on SearchLayouts in the Metadata API.</p>
DurableId	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Unique identifier for the field. Always retrieve this value before using it, as the value isn't guaranteed to stay the same from one release to the next. Use this field to simplify queries.</p>
EntityDefinition	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The name of the object associated with this search layout. Use in subqueries.</p>
EntityDefinitionId	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field	Details
	Description ID of the record in <code>EntityDefinition</code> . Use in subqueries.
<code>FieldsDisplayed</code>	Type SearchLayoutFieldsDisplayed Properties Nillable Description The list of fields displayed in a search result for the object. The name field is required. It's always displayed as the first column header, so it is not included in this list; all additional fields are included. The field name relative to the object name, for example <code>MyCustomField__c</code> , is specified for each custom field. This field is equivalent to the Search Results in the Search Layouts related list on the object detail page in the application user interface. It's also equivalent to <code>searchResultsAdditionalFields</code> in the Metadata API.
<code>Label</code>	Type string Properties Filter, Group, Nillable, Sort Description The label for this search layout.
<code>LayoutType</code>	Type string Properties Filter, Group, Nillable, Sort Description The type of search layout.

SearchLayoutButtonsDisplayed Metadata

Type	Details
<code>applicable</code>	Type boolean Description If <code>true</code> , the buttons listed in <code>buttons</code> apply to the object associated with this search layout.

Type	Details
buttons	Type string Description The buttons for which <code>applicable</code> is <code>true</code> .

SearchLayoutFieldsDisplayed Metadata

Type	Details
applicable	Type boolean Description If <code>true</code> , the fields listed in <code>fields</code> are available in the object associated with this search layout.
fields	Type string Description The list of fields on the object associated with this search layout.

SOQL Limitations

This object doesn't support some SOQL operations.

COUNT()

Example Query: `SELECT COUNT() FROM EntityDefinition`

Error Returned: The "DurableId" field is of type number, but the value from the external data source is "Account.SearchResult". Contact your administrator for help.

GROUP BY

Example Query: `SELECT COUNT(qualifiedapiname), isfeedenabled FROM EntityDefinition GROUP BY isfeedenabled`

Error Returned: The requested operation is not yet supported by this SObject storage type, contact salesforce.com support for more information.

LIMIT, LIMIT OFFSET

Example Queries:

`SELECT qualifiedapiname FROM EntityDefinition LIMIT 5`

`SELECT qualifiedapiname FROM EntityDefinition LIMIT 5 OFFSET 10`

An incorrect result is returned because LIMIT and LIMIT OFFSET are ignored.

NOT

Example Query: `SELECT qualifiedapiname FROM EntityDefinition WHERE qualifiedapiname!='Account'`

Error Returned: Only equals comparisons permitted

OR

Example Query: `SELECT qualifiedapiname, keyprefix FROM EntityDefinition WHERE isdeletable=true OR (isfeedenabled=false AND keyprefix='01j')`

Error Returned: Disjunctions not supported

ServiceFieldDataType

Don't use this object.

This object is visible in version 34.0 for some organizations, but we plan to remove it in a later release.

Scontrol

Represents a custom s-control, which is custom content that our system hosts, but client applications execute. An s-control can contain any type of content that you can display or run in a Web browser.



Important: Visualforce pages supersede s-controls. Organizations that haven't previously used s-controls can't create them.

Existing s-controls are unaffected, and can still be edited. We recommend that you move your s-controls to Visualforce. We continue to support this object.

Supported SOAP Calls

`query()`

Supported REST Methods


GET

Special Access Rules

- Your organization must be using Enterprise, Developer, or Unlimited Edition and be enabled for custom s-controls.
- Customer Portal users can't access this object.

Fields

Field	Details
ContentSource	Type picklist

Field	Details
	<p>Properties Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Specify the source of the s-control content, either custom HTML, a snippet (s-controls that are included in other s-controls), or a URL.</p>
Description	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort, Update</p> <p>Description Description of the custom s-control.</p>
DeveloperName	<p>Type string</p> <p>Properties Filter, Group, Sort, Update</p> <p>Description The unique name of the object in the API. This name can contain only underscores and alphanumeric characters, and must be unique in your organization. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores. In managed packages, this field prevents naming conflicts on package installations. With this field, a developer can change the object's name in a managed package and the changes are reflected in a subscriber's organization. Label is S-Control Name.</p> <p> Note: When creating large sets of data, always specify a unique <code>DeveloperName</code> for each record. If no <code>DeveloperName</code> is specified, Salesforce generates one for each record, which slows performance.</p>
EncodingKey	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort, Update</p> <p>Description Picklist of character set encodings, including ISO-08859-1, UTF-8, EUC, JIS, Shift-JIS, Korean (ks_c_5601-1987), Simplified Chinese (GB2312), and Traditional Chinese (Big5).</p>
Name	<p>Type string</p> <p>Properties Filter, Group, Sort, Update</p> <p>Description Required. Name of this custom s-control. Label is Label.</p>

Field	Details
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the <i>namespacePrefix__componentName</i> notation.</p> <p>The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are not Developer Edition organizations, <code>NamespacePrefix</code> is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.
SupportsCaching	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort, Update</p> <p>Description Indicates whether the s-control supports caching (<code>true</code>) or not (<code>false</code>).</p>

Usage

Use custom s-controls to manage custom content that extends application functionality. All users can view custom s-controls, but the “Customize Application” permission is required to create or update custom s-controls.

SOQLResult

A complex type that represents the result of a SOQL query in an `ApexExecutionOverlayResult` object. Available from API version 28.0 or later.

Fields

Field	Details
<code>queryError</code>	<p>Type string</p> <p>Description The error text returned if the execution was unsuccessful.</p>
<code>queryMetadata</code>	<p>Type QueryResultMetadata</p> <p>Description The structured result returned from a successful execution. QueryResultMetadata includes the following fields:</p> <ul style="list-style-type: none">• <code>columnMetadata</code>• <code>entityName</code>• <code>groupBy</code>• <code>idSelected</code>• <code>keyPrefix</code>
<code>queryResult</code>	<p>Type array of MapValue</p> <p>Description MapValue contains an array of MapEntry, which contains the following fields:</p> <ul style="list-style-type: none">• <code>keyDisplayValue</code>• <code>value</code> (reference to StateValue)

Usage

Overlay SOQL on checkpoints to capture structured debugging information. If your SOQL query may return more than one record when dealing with complex types, select only one row. For example, you can use a `LIMIT=1` clause in your SOQL query, or you can list rows for the user and have them select the row to inspect.

StandardAction

Represents the buttons, links, and actions (standard actions) for a standard or custom object. This object is available in API version 34.0 and later.

You can view the standard actions for an object by navigating to the Setup page and selecting **Buttons, Links, and Actions** for the relevant object.

Supported SOAP Calls

`query()`

Supported REST HTTP Methods

GET

Fields

Field	Details
<code>ContentType</code>	<p>Type string</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Indicates whether the button or link is standard, URL, s-control, JavaScript action, or Visualforce page. This value maps to the <code>Content Source</code> field in the user interface.</p>
<code>Description</code>	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Text displayed to an administrator in the standard action setup page. <code>Description</code> can be different from the <code>Label</code>, which is the label displayed in the user interface to end users. It can also be different from <code>Name</code>, which is a unique string used in merge fields.</p>
<code>DurableId</code>	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Unique identifier for the field. Always retrieve this value before using it, as the value isn't guaranteed to stay the same from one release to the next. <code>DurableId</code> in queries allows you to find the right record without having to retrieve the entire record.</p>
<code>EntityDefinition</code>	<p>Type EntityDefinition</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field	Details
	<p>Description</p> <p>The entity definition of the standard or custom object for which these standard actions are defined.</p> <p>For example, to find all the standard actions for Account, use a query similar to the following:</p> <pre>SELECT Label, EntityDefinition.Label FROM StandardAction WHERE EntityDefinition.QualifiedApiName = 'Account'</pre>
EntityDefinitionId	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>The ID of the standard or custom object for which this standard action is defined.</p>
IsOverriden	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Defaulted on create, Filter, Group, Sort</p> <p>Description</p> <p>Indicates whether this standard action has been overridden (<code>true</code>, or not).</p>
Label	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Nillable, Sort</p> <p>Description</p> <p>The text that displays in a user interface for the standard action.</p>
ManageableState	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Restricted picklist, Sort</p> <p>Description</p> <p>Indicates the action's manageable state.</p> <p>Valid values:</p> <ul style="list-style-type: none"> • beta • deleted • deprecated • installed

Field	Details
	<ul style="list-style-type: none"> released unmanaged
Name	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The unique name for the button or link when referenced from a merge field. This name can contain only underscores and alphanumeric characters, and must be unique in your organization. It must begin with a letter, not include spaces, not end with an underscore, and not contain two consecutive underscores.</p>
OverrideContent	<p>Type tns:Name</p> <p>Properties Filter, Group, Sort</p> <p>Description Reference to the actions that override standard actions for an object. To retrieve information, use this field with the fields in Name, because you can't query the field directly. For example, assume that you have overridden a standard action for Account. The following query returns the label of the standard action, and the name and ID of the action that overrides the standard action.</p> <pre>SELECT Label, OverrideContent.Id, OverrideContent.Name FROM StandardAction WHERE EntityDefinition.QualifiedApiName='Account' AND IsOverridden=true</pre> <p>Because OverrideContent is a Name object, you have access to all the fields in Name, in this case Name.Name and Name.Id.</p>
OverrideContentId	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of an OverrideContent record. Returns the same value as OverrideContent.Id in the sample SOQL query for OverrideContent.</p>

StaticResource

Represents the working copy of a static resource file for editing or saving. Static resources allow you to upload content that you can reference in a Visualforce page, including images, stylesheets, JavaScript, and other files. Available in Tooling API version 29.0 and later.

Supported SOAP API Calls

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

Supported REST API HTTP Methods

Query, GET, POST, PATCH, DELETE

Fields

Field Name	Details
Name	Type string Properties Create, Update Description The static resource name. The name can only contain characters, letters, and the underscore (_) character, must start with a letter, and cannot end with an underscore or contain two consecutive underscore characters
Body	Type string Properties Create, Update Description The data for the static resource file.
ContentType	Type string Properties Create, Update Description Required. The content type of the file, for example text/plain.
CacheControl	Type string Properties Create, Update

Field Name	Details
	Description Required. Indicates whether the static resource is marked with a public caching tag so that a third-party delivery client can cache the content. The valid values are: <ul style="list-style-type: none"> • Private • Public

Usage

To create, edit, or save a static resource file, create a `StaticResource` object that references it.

SymbolTable

A complex type that represents all user-defined tokens in the `Body` of an `ApexClass`, `ApexClassMember`, or `ApexTriggerMemeber` and their associated line and column locations within the `Body`.

Fields

Field	Details
<code>constructors</code>	Type array of <code>Constructor</code> Description Contains the position, scope, and signature of constructors for the Apex class. Apex triggers don't have constructors. Constructor includes the following fields: <ul style="list-style-type: none"> • <code>annotations</code> • <code>location</code> • <code>modifiers</code> • <code>name</code> • <code>references</code> • <code>visibility</code> (available only in API versions 33.0 and earlier; scope: Global, Public or Private) • <code>parameters</code>
<code>externalReferences</code>	Type array of <code>ExternalReference</code>

Field	Details
	<p>Description</p> <p>Contains the name, namespace, external class, method, and variable references for the Apex class or trigger. These references can be used for symbol highlighting or code navigation.</p> <p>ExternalReference includes the following fields:</p> <ul style="list-style-type: none"> • <code>methods</code> • <code>name</code> • <code>namespace</code> • <code>references</code> • <code>variables</code>
<code>innerClasses</code>	<p>Type</p> <p>array of SymbolTable</p> <p>Description</p> <p>Contains a symbol table for each inner class of the Apex class or trigger.</p>
<code>interfaces</code>	<p>Type</p> <p>array of String</p> <p>Description</p> <p>Contains a set of strings for each interface with the namespace and name, for example:</p> <pre>['System.Batchable', 'MyNamespace.MyInterface'].</pre>
<code>methods</code>	<p>Type</p> <p>array of Method</p> <p>Description</p> <p>Contains the position, name, scope, signature, and return type of available Apex methods.</p> <p>Method includes the following fields:</p> <ul style="list-style-type: none"> • <code>annotations</code> • <code>location</code> • <code>modifiers</code> • <code>name</code> • <code>references</code> • <code>visibility</code> (available only in API versions 33.0 and earlier; scope: Global, Public or Private) • <code>parameters</code> • <code>returnType</code>
<code>name</code>	<p>Type</p> <p>string</p> <p>Description</p> <p>The name of the Apex class or trigger.</p>

Field	Details
namespace	<p>Type string</p> <p>Description The namespace of the Apex class or trigger. Null if there is no namespace.</p>
parentClass	<p>Type string</p> <p>Description Returns parents of inner classes and extending classes.</p>
properties	<p>Type array of VisibilitySymbol</p> <p>Description Contains the position, name, scope, and references of properties for the Apex class or trigger. VisibilitySymbol includes the following fields:</p> <ul style="list-style-type: none"> • annotations • location • modifiers • name • references • visibility (available only in API versions 33.0 and earlier; scope: Global, Public or Private)
tableDeclaration	<p>Type array of Symbol</p> <p>Description Contains the position, name, and references of the Apex class or trigger. Symbol includes the following fields:</p> <ul style="list-style-type: none"> • annotations • location • modifiers • name • references
variables	<p>Type array of Symbol</p> <p>Description Contains the position, name, and references of related variables. Symbol includes the following fields:</p> <ul style="list-style-type: none"> • annotations

Field	Details
	<ul style="list-style-type: none"> • <code>location</code> • <code>modifiers</code> • <code>name</code> • <code>references</code>

Annotations

Available values for `annotations` fields include:

- `Deprecated`
- `Future`
- `HttpDelete`
- `HttpGet`
- `HttpPatch`
- `HttpPost`
- `HttpPut`
- `InvocableMethod`
- `InvocableVariable`
- `IsTest`
- `ReadOnly`
- `RemoteAction`
- `TestSetup`
- `TestVisible`
- `RestResource`

Modifiers

Modifiers can include more values than those values explicitly specified in classes and methods. All relevant modifiers, including implicit ones, are now returned. For example, all `webservice` methods have an implicit `global` modifier. Also, because fields and methods are private unless otherwise specified, the `private` modifier is returned by default.

The `testMethod` modifier is returned when either the `testMethod` modifier or the `IsTest` annotation is used.

Available values for `modifiers` fields include:

- `abstract`
- `final`
- `global`
- `override`
- `private`
- `protected`
- `public`

- `static`
- `testMethod`
- `transient`
- `virtual`
- `webService`
- `with sharing`
- `without sharing`

Usage

Use symbol tables instead of building a parser or compiler. Symbol tables allow you to do symbol highlighting, code navigation, code completion, symbol searches, and more.

A symbol table can't be created if the content referenced by the `ContentEntityId` field doesn't use a symbol table. Compiler errors for the last deployment of the [MetadataContainer](#) in the `MetadataContainerId` field also prevent a symbol table from being created.

TraceFlag

Represents a trace flag that triggers an Apex debug log at the specified logging level.

Supported SOAP API Calls

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

Supported REST API HTTP Methods

Query, GET, POST, PATCH, DELETE

Fields

Field Name	Details
ApexCode	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The log category level for Apex code. Includes information about Apex code and can include information such as log messages generated by data manipulation language (DML) statements, inline SOQL or SOSL queries, the start and completion of any triggers, the start and completion of any test method, and so on. The following are valid values.</p> <ul style="list-style-type: none"> • <code>Finest</code>

Field Name	Details
	<ul style="list-style-type: none">• <code>Finer</code>• <code>Fine</code>• <code>Debug</code>• <code>Info</code>• <code>Warn</code>• <code>Error</code> <p>This field is required.</p>
<code>ApexProfiling</code>	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The log category level for profiling information. Includes cumulative profiling information, such as the limits for your namespace, the number of emails sent, and so on. The following are valid values.</p> <ul style="list-style-type: none">• <code>Finest</code>• <code>Finer</code>• <code>Fine</code>• <code>Debug</code>• <code>Info</code>• <code>Warn</code>• <code>Error</code> <p>This field is required.</p>
<code>Callout</code>	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The log category level for callouts. Includes the request-response XML that the server is sending and receiving from an external Web service. This is useful when debugging issues related to SOAP API calls. The following are valid values.</p> <ul style="list-style-type: none">• <code>Finest</code>• <code>Finer</code>• <code>Fine</code>• <code>Debug</code>• <code>Info</code>• <code>Warn</code>• <code>Error</code>

Field Name	Details
	This field is required.
Database	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The log category for database activity. Includes information about database activity, including every DML statement or inline SOQL or SOSL query. The following are valid values.</p> <ul style="list-style-type: none"> • <code>Finest</code> • <code>Finer</code> • <code>Fine</code> • <code>Debug</code> • <code>Info</code> • <code>Warn</code> • <code>Error</code> <p>This field is required.</p>
ExpirationDate	<p>Type dateTime</p> <p>Properties Create, Filter, Sort, Update</p> <p>Description The date and time that the trace flag expires. This field is required.</p>
ScopeId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description A reference to a user. This field is used with the <code>TracedEntityID</code> field.</p> <ul style="list-style-type: none"> • When <code>ScopeId=user</code> the actions of the user/entity specified by <code>TracedEntityID</code> (user, Apex class or Apex trigger) are traced to the system log at the described level. System logs are visible only to you. Use this scope for class-level filtering. If there are both user and entity-level flags, the user flags take precedence until a method from a class with an entity trace flag is entered. When the method returns, the user trace flags are restored. • When <code>ScopeId=emptyid</code> the user's actions are traced to the organization's debug log at the described level. Debug logs are visible to all administrators. This option is only available if <code>TracedEntityID</code> references

Field Name	Details
	<p>a user (not an Apex class or Apex trigger). The variable <i>emptyid</i> can be the value 0000000000000000 or null.</p> <p>The scope defined here is reflected in the ApexLog Location field.</p>
System	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The log category level for calls to all system methods, such as the <code>System.debug</code> method. The following are valid values.</p> <ul style="list-style-type: none"> • <code>Finest</code> • <code>Finer</code> • <code>Fine</code> • <code>Debug</code> • <code>Info</code> • <code>Warn</code> • <code>Error</code> <p>This field is required.</p>
TracedEntityId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort, Update</p> <p>Description A reference to the following:</p> <ul style="list-style-type: none"> • Apex class • Apex trigger • User <p>This field is used with the <code>ScopeId</code> field. This field is required.</p>
Validation	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The log category level for validation rules. Includes information about validation rules, such as the name of the rule, whether the rule evaluated true or false, and so on. The following are valid values.</p> <ul style="list-style-type: none"> • <code>Finest</code>

Field Name	Details
	<ul style="list-style-type: none">• <code>Finer</code>• <code>Fine</code>• <code>Debug</code>• <code>Info</code>• <code>Warn</code>• <code>Error</code> <p>This field is required.</p>
Visualforce	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The log category level for Visualforce. Includes information about Visualforce events, including serialization and deserialization of the view state or the evaluation of a formula field in a Visualforce page. The following are valid values.</p> <ul style="list-style-type: none">• <code>Finest</code>• <code>Finer</code>• <code>Fine</code>• <code>Debug</code>• <code>Info</code>• <code>Warn</code>• <code>Error</code> <p>This field is required.</p>
Workflow	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort, Update</p> <p>Description The log category level for workflow rules. Includes information for workflow rules, such as the rule name, the actions taken, and so on. This field is required. The following are valid values.</p> <ul style="list-style-type: none">• <code>Finest</code>• <code>Finer</code>• <code>Fine</code>• <code>Debug</code>• <code>Info</code>• <code>Warn</code>• <code>Error</code>

Usage

To diagnose a functional issue or a performance problem, use the `TraceFlag` object to set up logging for yourself or for another user. The following options are available:

- To set up a log for a specific user, set `ScopeId` to null and `TracedEntityId` to the ID of the user. This option can only be configured for a user, not an Apex class or Apex trigger.
- To configure logging levels for system logs (visible only to you), set `ScopeId` to `user` and `TracedEntityId` to the ID of the logged-in user.
- To set up a system log (visible only to you) for a specific Apex class or trigger, set `ScopeId` to null and `TracedEntityId` to the ID of the Apex class or trigger.

The example below creates a new trace flag and attaches it to an end user.

```
//create a new TraceFlag object
TraceFlag traceFlag = new TraceFlag();
traceFlag.ApexCode = "Finest";
traceFlag.ApexProfiling = "Finest";
traceFlag.Callout = "Info";
traceFlag.Database = "Finest";
traceFlag.System = "Debug";
traceFlag.Validation = "Info";
traceFlag.Visualforce = "Info";
traceFlag.Workflow = "Info";

//set an expiration date
traceFlag.ExpirationDate = myTimestamp;
//set the ID of the user to monitor
traceFlag.TracedEntityId = "005A00000000i930";

//call the create method
TraceFlag[] traceFlags = { traceFlag };
SaveResult[] traceResults = sforce.create(traceFlags);
for (int i = 0; i < traceResults.Length; i++)
{
    if (traceResults[i].success)
    {
        Console.WriteLine("Successfully created trace flag: " +
            traceResults[i].id);
    }
    else
    {
        Console.WriteLine("Error: could not create trace flag ");
        Console.WriteLine("    The error reported was: " +
            traceResults[i].errors[0].message + "\n");
    }
}
```

User

Represents a user. You can retrieve standard fields on `User` with the Tooling API, but custom fields can't be retrieved.

This object is available in API version 32.0 and later.

Supported SOAP Calls

`describeLayout()`, `getDeleted()`, `getUpdated()`, `query()`, `retrieve()`, `search()`, `update()`

Supported REST HTTP Methods

GET, PATCH

Fields

Field	Details
FirstName	Type string Properties Create, Filter, Group, NillableSort, Update Description The user's first name.
LastName	Type string Properties Create, Filter, Group, NillableSort, Update Description The user's last name.
Name	Type string Properties Filter, Group, Sort Description Concatenation of <code>FirstName</code> and <code>LastName</code> . Limited to 121 characters.
Username	Type string Properties Create, Filter, Group, idLookup, Sort, Update Description The name of the user in your organization.
WorkspaceId	Type ID Properties Filter, Group, Nillable, Sort, Update

Field	Details
	Description The ID of the last open Developer Console workspace.

UserEntityAccess

Represents the access that the current user has to an object. Available in Tooling API version 34.0 and later.

Fields

Field	Details
DurableId	Type string Properties Filter, Group, Nillable, Sort Description Unique identifier for the field. Always retrieve this value before using it, as the value isn't guaranteed to stay the same from one release to the next. Use this field to simplify queries.
EntityDefinition	Type EntityDefinition Properties Filter, Group, Nillable, Sort Description The entity definition for the object associated with this user entity access record.
EntityDefinitionId	Type string Properties Filter, Group, Nillable, Sort Description ID of the EntityDefinition.
IsCreatable	Type boolean Properties Defaulted on create, Filter, Group, Sort Description If <code>true</code> , the user specified in the <code>User</code> field has access to create records of the associated object type.

Field	Details
IsDeletable	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, the user specified in the <code>User</code> field has access to delete records of the associated object type.</p>
IsEditable	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, the user specified in the <code>User</code> field has access to edit records of the associated object type.</p>
IsReadable	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, the user specified in the <code>User</code> field has access to view records of the associated object type.</p>
IsUndeleteable	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description If <code>true</code>, the user specified in the <code>User</code> field has access to undelete records of the associated object type.</p>
User	<p>Type User</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The user who has the access defined in this user entity access record, for the entity specified in the <code>EntityDefinition</code> field.</p>
UserId	<p>Type ID</p>

Field	Details
	Properties Filter, Group, Nillable, Sort
	Description ID of the user specified in the <code>User</code> field.

SOQL Limitations

This object doesn't support some SOQL operations.

COUNT()

Example Query: `SELECT COUNT() FROM EntityDefinition`

Error Returned: The "DurableId" field is of type number, but the value from the external data source is "Account.SearchResult". Contact your administrator for help.

GROUP BY

Example Query: `SELECT COUNT(qualifiedapiname), isfeedenabled FROM EntityDefinition GROUP BY isfeedenabled`

Error Returned: The requested operation is not yet supported by this SObject storage type, contact salesforce.com support for more information.

LIMIT, LIMIT OFFSET

Example Queries:

`SELECT qualifiedapiname FROM EntityDefinition LIMIT 5`

`SELECT qualifiedapiname FROM EntityDefinition LIMIT 5 OFFSET 10`

An incorrect result is returned because LIMIT and LIMIT OFFSET are ignored.

NOT

Example Query: `SELECT qualifiedapiname FROM EntityDefinition WHERE qualifiedapiname!='Account'`

Error Returned: Only equals comparisons permitted

OR

Example Query: `SELECT qualifiedapiname, keyprefix FROM EntityDefinition WHERE isdeletable=true OR (isfeedenabled=false AND keyprefix='01j')`

Error Returned: Disjunctions not supported

ValidationRule

Represents a validation rule or workflow rule which specifies the formula for when a condition is met. Available from API version 34.0 or later.

Supported SOAP Calls

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

Supported REST HTTP Methods

Query, GET, POST, PATCH

Fields

Field Name	Details
Active	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort.</p> <p>Description Required. Indicates whether this validation rule is active, (<code>true</code>), or not active (<code>false</code>).</p>
Description	<p>Type string</p> <p>Properties Filter, Nillable, Sort.</p> <p>Description A description of the validation rule.</p>
EntityDefinition	<p>Type EntityDefinition</p> <p>Properties Filter, Group, Sort.</p> <p>Description Required. The entity definition for the object associated with the validation rule.</p>
EntityDefinitionId	<p>Type string</p> <p>Properties Filter, Group, Sort.</p> <p>Description Required. ID of the record in <code>EntityDefinition</code>.</p>
ErrorDisplayField	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort.</p> <p>Description The fully specified name of a field in the application. If a value is supplied, the error message appears next to the specified field. If you do not specify a value or</p>

Field Name	Details
	the field isn't visible on the page layout, the value changes automatically to Top of Page.
ErrorMessage	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort .</p> <p>Description Required. The message that appears if the validation rule fails. The message must be 255 characters or less.</p>
FullName	<p>Type string</p> <p>Properties Create, Group, Nillable.</p> <p>Description The internal name of the object. White spaces and special characters are escaped for validity. The name must:</p> <ul style="list-style-type: none"> • Contain characters, letters, or the underscore (_) character • Must start with a letter • Can't end with an underscore • Can't contain two consecutive underscore characters. <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
Id	<p>Type Id</p> <p>Properties Defaulted on create, Filter, Group, idLookup, Sort.</p> <p>Description The unique system ID for this record.</p>
Metadata	<p>Type ValidationRule Metadata</p> <p>Properties Create, Nillable, Update.</p> <p>Description Validation rule metadata.</p> <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>

Field Name	Details
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort.</p> <p>Description The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix. Limit: 15 characters. You can refer to a component in a managed package by using the <i>namespacePrefix__componentName</i> notation.</p> <p>The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> • In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. • In organizations that are not Developer Edition organizations, <code>NamespacePrefix</code> is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.
ValidationName	<p>Type string</p> <p>Properties Filter, Group, Namefield, Sort.</p> <p>Description The name or ID of the object that this rule is associated with.</p>

ValidationRule Metadata

`active`, `description`, `errorDisplayField`, and `errorMessage` are described in the previous table.

Field Name	Details
errorConditionFormula	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Required. The formula defined in the validation rule. If the formula returns a value of <code>true</code>, an error message is displayed.</p>

WebLink

Represents a custom link to a URL or S-control. Available in the Tooling API from API version 34.0 or later.

Supported SOAP Calls

`getUpdated()`, `query()`, `retrieve()`, `search()`

Supported REST HTTP Methods

GET

Fields

Field Name	Details
Description	<p>Type string</p> <p>Properties Filter, Nillable, Sort</p> <p>Description A description of the WebLink.</p>
DisplayType	<p>Type string</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description Represents how this WebLink is rendered. Valid values:</p> <ul style="list-style-type: none">• <code>link</code> for a hyperlink• <code>button</code> for a button• <code>massAction</code> for a button attached to a related list
EncodingKey	<p>Type string</p> <p>Properties Filter, Sort</p> <p>Description Valid values include:</p> <ul style="list-style-type: none">• <code>UTF-8</code>—Unicode (UTF-8)• <code>ISO-8859-1</code>—General US & Western Europe (ISO-8859-1, ISO-LATIN-1)• <code>Shift_JIS</code>—Japanese (Shift-JIS)

Field Name	Details
	<ul style="list-style-type: none"> • ISO-2022-JP—Japanese (JIS) • EUC-JP—Japanese (EUC-JP) • x-SJIS_0213—Japanese (Shift-JIS_2004) • ks_c_5601-1987—Korean (ks_c_5601-1987) • Big5—Traditional Chinese (Big5) • GB2312—Simplified Chinese (GB2312) • Big5-HKSCS—Traditional Chinese Hong Kong (Big5-HKSCS)
EntityDefinition	<p>Type EntityDefinition</p> <p>Properties Filter, Group, Sort.</p> <p>Description Required. Available in version 34.0. The entity definition for the object associated with this WebLink.</p>
EntityDefinitionId	<p>Type string</p> <p>Properties Filter, Group, Sort.</p> <p>Description Required. ID of the record associated with this WebLink. The record's object type is in <code>EntityDefinition</code>.</p>
FullName	<p>Type string</p> <p>Properties Filter, Group, Sort.</p> <p>Description The full name of the associated metadata object in Metadata API. Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
HasMenubar	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort.</p> <p>Description If <code>OpenType</code> is <code>newWindow</code>, whether to show the browser menu bar for the popup window (<code>true</code>, or not (<code>false</code>)). For other values of <code>OpenType</code>, don't specify a value here.</p>

Field Name	Details
HasScrollbars	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort.</p> <p>Description If the value of <code>OpenType</code> is <code>newWindow</code>, whether to show the scroll bars for the window (<code>true</code>) or not (<code>false</code>). For other values of <code>OpenType</code>, don't specify a value here.</p>
HasToolbar	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort.</p> <p>Description If the value of <code>OpenType</code> is <code>newWindow</code>, whether to show the browser toolbar for the window (<code>true</code>) or not (<code>false</code>). For other values of <code>OpenType</code>, don't specify a value here.</p>
Height	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort.</p> <p>Description Required if the value of <code>OpenType</code> is <code>newWindow</code>. Height in pixels of the window opened by this <code>WebLink</code>. For other values of <code>OpenType</code>, don't specify a value here.</p>
IsResizable	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort.</p> <p>Description If the value of <code>OpenType</code> is <code>newWindow</code>, whether to allow resizing of the window (<code>true</code>) or not (<code>false</code>). For other values of <code>OpenType</code>, don't specify a value here.</p>
LinkType	<p>Type string</p> <p>Properties Filter, Group, Restricted picklist, Sort</p>

Field Name	Details
	<p>Description</p> <p>Required. Represents whether the content of this WebLink is specified by a URL, an sControl, a JavaScript code block, or a Visualforce page.</p> <ul style="list-style-type: none"> • url • sControl • javascript • page • flow—Reserved for future use.
MasterLabel	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Nillable, Sort, Update</p> <p>Description</p> <p>Master label for this object. This display value is the internal label that is not translated. Limit: 240 characters.</p>
Metadata	<p>Type</p> <p>mns:WebLink</p> <p>Properties</p> <p>Filter, Group, idLookup, Sort</p> <p>Description</p> <p>The metadata for this object as defined in the Metadata API.</p> <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
Name	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, idLookup, Sort</p> <p>Description</p> <p>Required. Name to display on the page.</p>
NamespacePrefix	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Sort.</p> <p>Description</p> <p>The namespace prefix associated with this object. Each Developer Edition organization that creates a managed package has a unique namespace prefix.</p>

Field Name	Details
	<p>Limit: 15 characters. You can refer to a component in a managed package by using the <code>namespacePrefix__componentName</code> notation.</p> <p>The namespace prefix can have one of the following values:</p> <ul style="list-style-type: none"> In Developer Edition organizations, the namespace prefix is set to the namespace prefix of the organization for all objects that support it. There is an exception if an object is in an installed managed package. In that case, the object has the namespace prefix of the installed managed package. This field's value is the namespace prefix of the Developer Edition organization of the package developer. In organizations that are not Developer Edition organizations, <code>NamespacePrefix</code> is only set for objects that are part of an installed managed package. There is no namespace prefix for all other objects.
OpenType	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description Valid values:</p> <ul style="list-style-type: none"> <code>newWindow</code> <code>sidebar</code> <code>noSidebar</code> <code>replace</code> <code>onClickJavaScript</code>
Position	<p>Type string</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description If the value of <code>OpenType</code> is <code>newWindow</code>, how the new window should be displayed. Otherwise, don't specify a value. Valid values:</p> <ul style="list-style-type: none"> <code>fullScreen</code> <code>none</code> <code>topLeft</code>
RequireRowSelection	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p>

Field Name	Details
	<p>Description</p> <p>If the value of <code>OpenType</code> is <code>massAction</code>, indicates whether to require individual row selection to execute the action for this button (<code>true</code>) or not (<code>false</code>). Otherwise, leave this field empty.</p>
<code>Scontrol</code>	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>If the value of <code>LinkType</code> is <code>sControl</code>, this field represents the name of the <code>sControl</code>. Otherwise, leave this field empty.</p>
<code>ShowsLocation</code>	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>If the value of <code>OpenType</code> is <code>newWindow</code>, indicates whether to show the browser location bar for the window (<code>true</code>) or not (<code>false</code>). Otherwise, leave this field empty.</p>
<code>ShowsStatus</code>	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>If the value of <code>OpenType</code> is <code>newWindow</code>, show the browser status bar for the window (<code>true</code>). Otherwise, don't specify a value.</p>
<code>Url</code>	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Nillable, Sort</p> <p>Description</p> <p>If the value of <code>LinkType</code> is <code>url</code>, <code>Url</code> represents the URL value. If the value of <code>LinkType</code> is <code>javascript</code>, <code>Url</code> represents the JavaScript content. For other values of <code>LinkType</code>, leave this field empty.</p> <p>Content must be escaped in a manner consistent with XML parsing rules.</p> <p>Required. URL of the page to link to. Can include fields as tokens within the URL. Limit: 1,024 KB.</p>

Field Name	Details
Width	Type int Properties Filter, Group, Nillable, Sort Description Width in pixels of the window opened by this Weblink. Required if the value of <code>OpenType</code> is <code>newWindow</code> . Otherwise, leave this field empty.

WorkflowAlert

Represents a workflow alert. A workflow alert is an email generated by a workflow rule or approval process and sent to designated recipients.

This object is available in API version 32.0 and later.

Supported SOAP Calls

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

Supported REST HTTP Methods

DELETE, GET, PATCH, POST

Fields

Field	Details
CcEmails	Type string Properties Filter, Nillable, Sort Description Additional CC email addresses.
Description	Type string Properties Filter, Group, idLookup, Sort Description A description of the workflow alert.

Field	Details
DeveloperName	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The unique name of the workflow alert in the API.</p>
EntityDefinitionId	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the entity containing the alert.</p>
FullName	<p>Type string</p> <p>Properties Create, Group, Nillable</p> <p>Description The full name of the associated metadata object in Metadata API. Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
Metadata	<p>Type WorkflowAlertMetadata</p> <p>Properties Create, Nillable, Update</p> <p>Description Alert definition metadata. Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
NamespacePrefix	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The namespace of the package to uniquely identify the workflow alert.</p>

Field	Details
SenderType	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Restricted picklist, Sort</p> <p>Description The type of sender. Values are:</p> <ul style="list-style-type: none">• CurrentUser• OrgWideEmailAddress• DefaultWorkflowUser
TemplateId	<p>Type ID</p> <p>Properties Filter, Group, Sort</p> <p>Description A reference to an email template.</p>

WorkflowFieldUpdate

Represents a workflow field update.

This object is available in API version 32.0 and later.

Supported SOAP Calls

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

Supported REST HTTP Methods

DELETE, GET, PATCH, POST

Fields

Field	Details
EntityDefinitionId	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the entity containing the workflow field update.</p>

Field	Details
FieldDefinitionId	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the field for the workflow field update.</p>
FullName	<p>Type string</p> <p>Properties Create, Group, Nillable</p> <p>Description The full name of the associated metadata object in Metadata API. Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
LiteralValue	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description If the update uses a literal value, this is that value.</p>
LookupValueId	<p>Type ID</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description If the update looks up a value, this is the lookup value referenced.</p>
Metadata	<p>Type WorkflowFieldUpdateMetadata</p> <p>Properties Create, Nillable, Update</p> <p>Description The workflow field update metadata. Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>

Field	Details
Name	Type string Properties Filter, Group, idLookup, Sort Description The name of the workflow field update.
NamespacePrefix	Type string Properties Filter, Group, Nillable, Sort Description The namespace of the package containing the workflow field update object.
SourceTableEnumOrId	Type picklist Properties Filter, Group, Restricted picklist, Sort Description The enum (for example, Account) or ID of the object this workflow field update is on.

WorkflowOutboundMessage

Represents an outbound message. An outbound message is a workflow, approval, or milestone action that sends the information you specify to an endpoint you designate, such as an external service. Outbound messaging is configured in the Salesforce setup menu. Then you must configure the external endpoint. You can create a listener for the messages using the SOAP API.

This object is available in API version 32.0 and later.

Supported SOAP Calls

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

Supported REST HTTP Methods

DELETE, GET, PATCH, POST

Fields

Field	Details
ApiVersion	<p>Type double</p> <p>Properties Filter, Sort</p> <p>Description The API version is automatically generated and set to the current API version when the outbound message was created.</p>
EntityDefinitionId	<p>Type string</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the entity containing the outbound message.</p>
FullName	<p>Type string</p> <p>Properties Create, Group, Nillable</p> <p>Description The full name of the associated metadata object in Metadata API. Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
IntegrationUserId	<p>Type ID</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the user under which this message is sent.</p>
Metadata	<p>Type WorkflowOutboundMessageMetadata</p> <p>Properties Create, Nillable, Update</p> <p>Description Outbound message definition metadata.</p>

Field	Details
	Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.
Name	Type string Properties Filter, Group, idLookup, Sort Description The name of the outbound message.
NamespacePrefix	Type string Properties Filter, Group, Nillable, Sort Description The namespace of the package containing the outbound message.

WorkflowRule

Represents a workflow rule that is used to fire off a specific workflow action when the specified criteria is met. Includes access to the associated WorkflowRule object in Salesforce Metadata API.

Available from API version 30.0 or later.

Supported SOAP Calls

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

Supported REST HTTP Methods

Query, DELETE, GET, PATCH, POST

Fields

Field Name	Details
FullName	Type string Properties Create, Group, Nillable

Field Name	Details
	<p>Description</p> <p>The full name of the associated metadata object in Metadata API.</p> <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
Metadata	<p>Type</p> <p>WorkflowRuleMetadata</p> <p>Properties</p> <p>Create, Nillable, Update</p> <p>Description</p> <p>Workflow rule metadata.</p> <p>Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.</p>
Name	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>The enum name or ID of entity this rule is associated with.</p>
NamespacePrefix	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, idLookup, Sort</p> <p>Description</p> <p>The namespace of the package containing the workflow rule object.</p>
TableEnumOrId	<p>Type</p> <p>picklist</p> <p>Properties</p> <p>Filter, Group, Restricted picklist, Sort</p> <p>Description</p> <p>The enum (for example, Account) or ID of the object for this workflow rule.</p>

WorkflowTask

Represents a workflow task that is used to fire off a specific workflow action when the specified criteria is met. Includes access to the associated WorkflowRule object in Salesforce Metadata API.

Available from API version 32.0 or later.

Supported SOAP Calls

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

Supported REST HTTP Methods

Query, DELETE, GET, PATCH, POST

Fields

Field Name	Details
EntityDefinitionId	Type string Properties Filter, Group, Sort Description The ID of the entity containing the workflow task.
FullName	Type string Properties Create, Group, Nillable Description The full name of the associated metadata object in Metadata API. Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.
Metadata	Type WorkflowTaskMetadata Properties Create, Nillable, Update Description Workflow task metadata. Query this field only if the query result contains no more than one record. Otherwise, an error is returned. If more than one record exists, use multiple queries to retrieve the records. This limit protects performance.
NamespacePrefix	Type string

Field Name	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description The namespace of the package containing the workflow task object.</p>
Priority	<p>Type picklist</p> <p>Properties Filter, Group, Sort</p> <p>Description The task's priority. Values are:</p> <ul style="list-style-type: none">• High• Normal• Low
Status	<p>Type picklist</p> <p>Properties Filter, Group, Sort</p> <p>Description The task's status. Values are:</p> <ul style="list-style-type: none">• Not Started• In Progress• Completed• Waiting on someone else• Deferred
Subject	<p>Type string</p> <p>Properties Filter, Group, idLookup, Sort</p> <p>Description A subject for the workflow task. It is used if an email notification is sent when the task is assigned.</p>

CHAPTER 3 AllOrNoneHeader

Enables the rollback of all record changes when some of the records in a call result in failures. When this header is set to `true`, changes are saved only when all records are processed successfully.

Version

This header is available in API version 34.0 and later.

Supported Calls

`create()`, `update()`, `upsert()`, `delete()`

The `AllOrNoneHeader` header is also supported by Tooling REST API resources that create, update, or delete records. See [Use Tooling API with REST](#).

Usage

By default, if this header isn't used, a call can save a partial set of records—the records that are processed successfully are saved and records that have failures aren't saved.

Fields

Field Name	Type	Description
<code>allOrNone</code>	boolean	Set to <code>true</code> to cause all record changes to be rolled back if any records in the call have failures. Set to <code>false</code> to enable saving the successful record changes when some records in the call cause failures.

Sample Code—Java

When using REST, add the `AllOrNoneHeader` to your `HttpRequest` instance before you perform a call as follows:

```
HttpRequest req = new HttpRequest();
req.setHeader("AllOrNoneHeader", true);
```

When using SOAP, add the `AllOrNoneHeader` to the tooling connection before you perform a call as follows:

```
toolingConnection.setAllOrNoneHeader(true);
```

This next example shows how to use the `AllOrNoneHeader` when creating two Apex classes with the Tooling SOAP API. Because the second class doesn't have the required `Body` field, the `create()` call can't create this class and rolls back the first class. The output is shown after this code sample.

```
import com.sforce.soap.tooling.ApexClass;
import com.sforce.soap.tooling.CustomObject;
import com.sforce.soap.tooling.DeploymentStatus;
import com.sforce.soap.tooling.SaveResult;
import com.sforce.soap.tooling.SoapConnection;
import com.sforce.ws.ConnectionException;

public class HeaderSample {
    SoapConnection toolingConnection = null;

    public static void main(String[] args) throws ConnectionException
    {
        HeaderSample samples = new HeaderSample();
        samples.createWithHeader();
    }

    public HeaderSample() throws ConnectionException {
        toolingConnection = ToolingLoginUtil.login();
    }

    public void createWithHeader() throws ConnectionException {
        String classBody1 = "public class MyFirstClass {\n"
            + "    public string SayHello() {\n"
            + "        return 'Hello';\n"
            + "    }\n"
            + "}";

        // Create two new Apex Class objects
        ApexClass apexClass1 = new ApexClass();
        apexClass1.setBody(classBody1);
        ApexClass apexClass2 = new ApexClass();
        // No class body for class2

        ApexClass[] classes = { apexClass1, apexClass2 };

        // Setting the allOrNone header to true to cause
        // the call to not commit any record if one or more
        // records in this call have failures.
        toolingConnection.setAllOrNoneHeader(true);

        // Call create() to add the Apex classes
        SaveResult[] saveResults = toolingConnection.create(classes);

        for (int i = 0; i < saveResults.length; i++) {
            if (saveResults[i].isSuccess()) {
                System.out.println("Successfully created class.");
            } else {

```

```

        System.out.println("Error: could not create class.");

        System.out.println("    The error message is: " +
            saveResults[i].getErrors()[0].getMessage());
        System.out.println("    The error status code is: " +

            saveResults[i].getErrors()[0].getStatusCode()
+ "\n");
    }
}
}
}

```

This is the output that the sample returns. The first record is rolled back and the second has a failure.

```

Error: could not create class
    The error message is: Record rolled back because not all records
were valid and the request was using AllOrNone header
    The error status code is: ALL_OR_NONE_OPERATION_ROLLED_BACK

Error: could not create class
    The error message is: Empty script
    The error status code is: INVALID_FIELD_FOR_INSERT_UPDATE

```

INDEX

A

AllOrNoneHeader header [215](#)
Apex
 Debugging [32, 34, 37, 186](#)
 Deploying [68, 144](#)
 Editing [22–23, 40, 50, 54](#)
 Saving [22, 40, 50](#)
 Saving and compiling [23, 54, 68, 144](#)
 Viewing code coverage [26, 28, 39](#)
ApexClass object [22](#)
ApexClassMember object [23](#)
ApexCodeCoverage object [26](#)
ApexCodeCoverageAggregate object [28](#)
ApexComponent object [29](#)
ApexComponentMember object [30](#)
ApexExecutionOverlayAction object [32](#)
ApexExecutionOverlayResult object [34](#)
ApexLog object [37](#)
ApexOrgWideCoverage object [39](#)
ApexPage object [40](#)
ApexPageMember object [40](#)
ApexResult object [43](#)
ApexTestQueueItem object [43](#)
ApexTestResult object [48](#)
ApexTrigger object [50](#)
ApexTriggerMember object [54](#)
AuraDefinition object [56, 58](#)

B

BusinessProcess object [60](#)

C

Checkpoint [43, 133, 176](#)
CompactLayout object [61](#)
CompactLayoutInfo object [64](#)
CompactLayoutItemInfo object [62](#)
Compile errors [80, 82, 147, 150, 165, 168, 193](#)
ContainerAsyncRequest object [68](#)
CreatedById fields [12](#)
CreatedDate fields [12](#)
CSS
 Editing [181](#)
CustomField object [70](#)
CustomFieldMember object [72](#)
CustomObject object [74](#)

CustomTab object [77](#)

D

DataType object [80](#)
Debugging [37, 186](#)
Debugging Apex [32, 34](#)
DeployDetails object [82](#)
Deploying Apex [68, 144](#)
Deploying Visualforce [68, 144](#)
Developer Console [1](#)

E

Editing Apex [22–23, 40, 50, 54](#)
Editing Visualforce [29–30, 40](#)
EmailTemplate object [83](#)
EntityDefinition object [85](#)
EntityLimit object [96](#)
EntityParticle object [98](#)

F

FieldDefinition object [105](#)
Fields
 system fields [12](#)
FieldSet object [124](#)
FlexiPage object [125](#)
Flow object [127](#)
FlowDefinition object [130](#)

G

Generating heap dumps [32, 34, 37](#)

H

Headers
 AllOrNoneHeader [215](#)
Heap dump [133](#)
Heap dumps [32, 34, 37](#)
HeapDump object [133](#)
HistoryRetentionJob object [134](#)

I

ID fields [12](#)

J

JavaScript
 Editing [181](#)

L

LastModifiedById fields [12](#)
 LastModifiedDate fields [12](#)
 Layout object [136](#)
 Lightning components [56](#), [58](#)
 Log [43](#), [133](#), [176](#)
 Logging [37](#), [186](#)
 LookupFilter object [138](#)

M

MenuItem object [141](#)
 MetadataContainer object [144](#)

O

Objects

ApexClass [22](#)
 ApexClassMember [23](#)
 ApexCodeCoverage [26](#)
 ApexCodeCoverageAggregate [28](#)
 ApexComponent [29](#)
 ApexComponentMember [30](#)
 ApexExecutionOverlayAction [32](#)
 ApexExecutionOverlayResult [34](#)
 ApexLog [37](#)
 ApexOrgWideCoverage [39](#)
 ApexPage [40](#)
 ApexPageMember [40](#)
 ApexResult [43](#)
 ApexTestQueueItem [43](#)
 ApexTestResult [48](#)
 ApexTrigger [50](#)
 ApexTriggerMember [54](#)
 AuraDefinition [56](#), [58](#)
 BusinessProcess [60](#), [177](#)
 CompactLayout [61](#)
 CompactLayoutInfo [64](#)
 CompactLayoutItemInfo [62](#)
 ContainerAsyncRequest [68](#)
 CustomField [70](#)
 CustomFieldMember [72](#)
 CustomObject [74](#)
 CustomTab [77](#)
 DataType [80](#)
 DeployDetails [82](#)
 EmailTemplate [83](#)
 EntityDefinition [85](#)
 EntityLimit [96](#)
 EntityParticle [98](#)
 FieldDefinition [105](#)

Objects (*continued*)

FieldSet [124](#)
 FlexiPage [125](#)
 Flow [127](#)
 FlowDefinition [130](#)
 HeapDump [133](#)
 HistoryRetentionJob [134](#)
 Layout [136](#)
 LookupFilter [138](#)
 MenuItem [141](#)
 MetadataContainer [144](#)
 process flows [127](#)
 Profile [145](#)
 ProfileLayout [146](#)
 Publisher [147](#)
 QueryResult [150](#)
 QuickActionDefinition [151](#)
 QuickActionList [156](#)
 QuickActionListItem [157](#)
 RecentlyViewed [158](#)
 RecordType [163](#)
 RelationshipDomain [165](#)
 RelationshipInfo [168](#)
 Scontrol [174](#)
 SearchLayout [170](#)
 ServiceFieldDataType [174](#)
 SOQLResult [176](#)
 StaticResource [181](#)
 SymbolTable [182](#)
 TraceFlag [186](#)
 User [191](#)
 UserEntityAccess [193](#)
 ValidationRule [195](#)
 WebLink [199](#)
 WorkflowAlert [205](#)
 WorkflowFieldUpdate [207](#)
 WorkflowOutboundMessage [209](#)
 WorkflowRule [211](#)
 WorkflowTask [212](#)

Overview [1](#)

P

process flows [130](#)
 Profile object [145](#)
 ProfileLayout object [146](#)
 Publisher object [147](#)

Q

QueryResult object [150](#)

QuickActionDefinition object [151](#)
QuickActionList object [156](#)
QuickActionListItem object [157](#)

R

RecentlyViewed object [158](#)
RecordType object [163](#)
RelationshipDomain object [165](#)
RelationshipInfo object [168](#)
REST API [2](#)

S

Saving and compiling Apex [23](#), [54](#), [68](#), [144](#)
Saving and compiling Visualforce [30](#), [40](#), [68](#), [144](#)
Scontrol object [174](#)
SearchLayout object [170](#)
ServiceFieldDataType object [174](#)
SOAP API [5](#)
SOQLResult object [176](#)
Standard objects [11](#), [13–14](#), [16–18](#)
StandardAction object [177](#)
StaticResource object [181](#)
Symbol tables [182](#)
SymbolTable object [182](#)
System fields [12](#)
SystemModstamp fields [12](#)

T

Tasks [1](#)
Tests [43](#), [48](#)
TraceFlag object [186](#)

U

User object [191](#)
UserEntityAccess object [193](#)

V

ValidationRule object [195](#)
Visualforce
 Deploying [68](#), [144](#)
 Editing [29–30](#), [40](#)
 Saving and compiling [29–30](#), [40](#), [68](#), [144](#)

W

WebLink object [199](#)
WorkflowAlert object [205](#)
WorkflowFieldUpdate object [207](#)
WorkflowOutboundMessage object [209](#)
WorkflowRule object [211](#)
WorkflowTask object [212](#)

X

XML
 Editing [181](#)