

salesforce

Salesforce Knowledge Developer Guide

Version 55.0, Summer '22



 @salesforcedocs

Last updated: August 3, 2022

CONTENTS

Chapter 1: Overview	1
Knowledge Object Model	2
Knowledge Articles and Data Categories in the API	5
API End-of-Life	8
Chapter 2: SOAP API	9
Objects	10
KnowledgeArticle	11
KnowledgeArticleVersion	13
Knowledge__DataCategorySelection	24
Knowledge__Feed	25
Knowledge__kav	30
Knowledge__ka	38
KnowledgeArticleVersionHistory	41
KnowledgeArticleViewStat	43
KnowledgeArticleVoteStat	45
CaseArticle	46
LinkedArticle	48
RecentlyViewed	50
SearchPromotionRule	54
TopicAssignment	55
Calls	57
describeKnowledge()	58
describeDataCategoryGroups()	59
describeDataCategoryGroupStructures()	61
search()	66
Chapter 3: REST API	73
Invocable Actions	74
Knowledge Actions	74
Manage Knowledge	85
Archive the Master Version of an Article	86
Assign a Task Related to a Master Article	87
Assign a Task Related to a Translation	88
Delete a Master Version of an Article	89
Delete a Translated Version of an Article	89
Edit an Online Version of a Master Article	90
Get a List of Salesforce Site and Experience Cloud Site URLs for an Online Article	90
Publish the Master Version of an Article	91

Contents

Restore an Archived Version of an Article	92
Retrieve Article Metadata	93
Retrieve Knowledge Language Settings	93
Retrieve a Version of an Article	94
Search for Metadata Elements of a Master Version	95
Search for Metadata Elements of a Translated Version	95
Set a Translated Article Version to Complete	96
Set a Translated Article Version to Incomplete	97
Submit an Article for Translation	97
Unpublish the Master Version of an Article	98
Unpublish the Online Version of a Translated Article	99
Parameterized Search	100
Search	108
Search Scope and Order	108
Search Result Layouts	109
Search for Records Suggested by Autocomplete and Instant Results	110
Search Suggested Article Title Matches	116
Search Suggested Queries	118
Support Knowledge	120
Data Category Groups	121
Data Category Detail	123
Articles List	125
Articles Details	129
Chapter 4: Metadata API	135
ArticleType	136
ArticleType Layout	139
ChannelLayout	141
ArticleType CustomField	142
KnowledgeSettings	145
DataCategoryGroup	152
SearchSettings	158
SearchLayouts	162
SynonymDictionary	165
ExternalDataSource	167
Chapter 5: SOQL and SOSL	174
Update an Article's Keyword Tracking with SOQL	176
Update an Article Viewstat with SOQL	176
WITH filteringExpression	176
WITH DATA CATEGORY filteringExpression	178
Using Relationship Queries with Data Category Selection Objects	181
About SOSL	182
Update an Article's Keyword Tracking with SOSL	183

Contents

Update an Article's Viewstat with SOSL	184
WITH DATA CATEGORY DataCategorySpec	184
WITH SNIPPET	186
Chapter 6: Apex	189
PublishingService Class	190
PublishingService Methods	190
INDEX	202

CHAPTER 1 Developing with Salesforce Knowledge

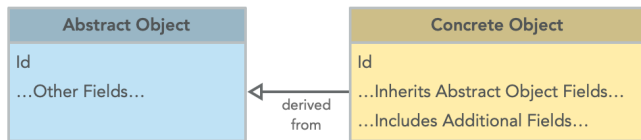
In this chapter ...

- [Knowledge Object Model](#)
- [Knowledge Articles and Data Categories in the API](#)
- [API End-of-Life](#)

Salesforce Knowledge gives your website visitors, clients, partners, and service agents the ultimate in support. You can create and manage your company information and securely share it when and where it is needed. Most features are declarative and let you point-and-click your way to a successful implementation. However, the API allows you to code your way to case deflection, happier customers, and more productive support agents. This guide discusses both Lightning Knowledge and Knowledge in Salesforce Classic.

Knowledge Object Model

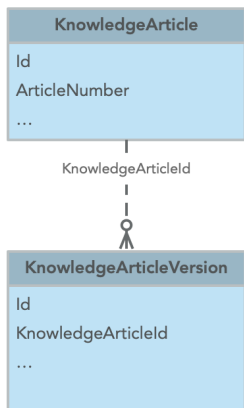
The Knowledge object model is slightly different from other feature areas because it exposes a set of abstract Salesforce objects that aren't directly used when you create articles. These abstract objects then contain concrete derivations that you do use when creating articles.



This architecture gives more flexibility to search behavior in support of Lightning Knowledge and Salesforce Classic.

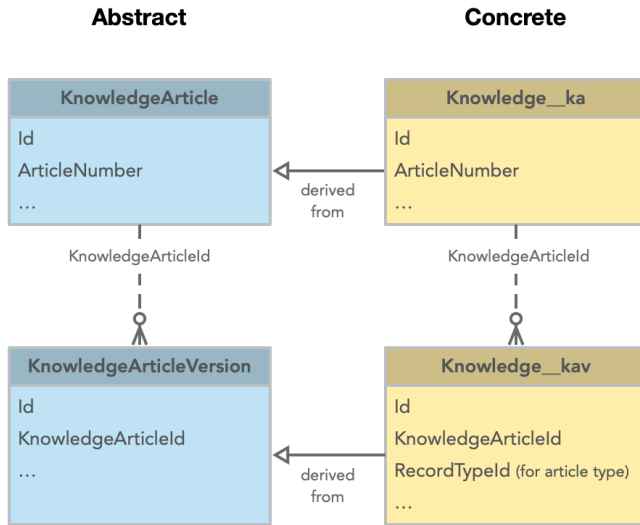
Core Knowledge Objects

The core abstract Knowledge objects are for a Knowledge article (`KnowledgeArticle`), and a set of versions of that article (`KnowledgeArticleVersion`). These versions can include different translations and historical versions of an article.



- [KnowledgeArticle](#)—The abstract Knowledge article object. This object contains metadata about a knowledge article irrespective of the article's version, translation, or status.
- [KnowledgeArticleVersion](#)—The abstract Knowledge article version object. Whenever you create a draft of an article, it gets a new version number and a unique ID. Each translation of an article also has a version and ID. Each version is a `KnowledgeArticleVersion` record with the `KnowledgeArticle` record as the parent.

When you create knowledge articles, you create records with concrete Salesforce objects. These objects inherit from their abstract counterparts. By default, these concrete objects are named “`Knowledge__ka`” for knowledge articles and “`Knowledge__kav`” for knowledge article versions in Lightning Knowledge.



This diagram shows the default names in Lightning Knowledge for the concrete implementations, but you can change these names. The names always have the suffixes “__ka” for knowledge articles and “__kav” for knowledge article versions. You can change the “Knowledge” prefix by changing the Object Name for the Knowledge__kav object in the Object Manager.

The concrete implementation contains any of your custom fields for a knowledge article. The Knowledge__kav object also contains a “RecordTypeId” field, which is a record type used to describe the article structure, such as FAQ or Tutorial. Each record type can contain its own layouts. For example, an FAQ record type can use a special layout that displays custom fields such as “Question__c” and “Answer__c”.

In Salesforce Classic, the article structure is determined by the article type. Each knowledge article type has a different concrete implementation, for example FAQ__kav, Tutorial__kav. In Lightning Knowledge, one concrete object is used, for example, Knowledge__kav, and instead the record type is used to associate an article with a unique article structure.

Complete Knowledge Object Diagram

To see the relationship of Knowledge articles to other objects, let’s highlight some of the other important object relationships. In this diagram, the default value “Knowledge” is the article type name to illustrate the object model. If you change this value in Setup, the prefix for each object with “__” in the name is changed accordingly.

- [KnowledgeArticleVersion](#) on page 13—The abstract Knowledge article version object. Whenever you create a draft of an article, it gets a new version number and a unique ID. Each translation of an article also has a version and ID. Each version is a [KnowledgeArticleVersion](#) on page 13 record with the [KnowledgeArticle](#) on page 11 record as the parent.
- [KnowledgeArticleVersionHistory](#)—Enables read-only access to the full history of an article.
- [KnowledgeArticleViewStat](#)—Provides statistics on the number of views for the specified article.
- [KnowledgeArticleVoteStat](#)—Provides the weighted rating for the specified article.
- [LinkedArticle](#)—A knowledge article that is attached to a work order, work order line item, or work type.
- [LinkedArticleFeed](#)—The comment feed on a linked article.
- [LinkedArticleHistory](#)—The history of changes made to tracked fields on a linked article.
- [WorkOrder](#)—Field service work to be performed for a customer.
- [WorkOrderLineItem](#)—A subtask on a work order in field service.
- [Vote](#)—A vote that a user has made on a knowledge article.
 - Using Vote is allowed for Knowledge Article objects but not for Knowledge Article Version objects.
 - Only one vote record can exist per knowledge article per user. Voting isn't allowed by Site Guest Users.
 - Vote allows multiple types of values for the "Type" field, but only "Up" and "Down" are allowed for Knowledge Articles.

Knowledge Articles and Data Categories in the API

Before working with the Knowledge API, learn the basics about knowledge articles and data categories.

Articles Overview

Articles capture information about your company's products and services that you want to make available in your knowledge base. Articles in the knowledge base can be classified using data categories to make it easy for users to find the articles they need. Administrators can use data categories to control access to articles.

Knowledge Articles vs. Knowledge Article Versions

When working with articles, keep in mind that the [KnowledgeArticle](#) represents the parent record of all article versions. [KnowledgeArticleVersion](#) records represent each version of a given article.

Record Types vs. Article Types

The article structure is represented differently between Lightning Experience and Salesforce Classic. In Lightning Knowledge, you use the same record types available in other custom objects (see the `RecordTypeId` field on [Knowledge__kav](#)) to structure different types of articles. For example, you can use different layouts for different record types. In Salesforce Classic, you get this functionality through article types (see the `ArticleType` field on [KnowledgeArticleVersion](#)). Each unique type of article has a unique object in Salesforce Classic (for example, `FAQ__kav` for FAQ article types). Lightning Knowledge does not have a unique object for each type because it is handled using the record type.

Audience Channel

An audience, sometimes called a channel, refers to the types of users who can access an article. Salesforce Knowledge offers four channels where you can make articles available.

- Internal App: Salesforce users can access articles depending on their role visibility.
- Customer: Customers can access articles in a community, site, or customer portal. Customer users inherit the role visibility of the manager on the account. In a community, the article is only available to users with Customer Community or Customer Community Plus licenses.

- **Partner:** Partners can access articles in a community, site, or partner portal. Partner users inherit the role visibility of the manager on the account. In a community, the article is only available to users with Partner Community licenses.
- **Public Knowledge Base:** Articles can be made available to anonymous users by creating a public knowledge base. With Lightning Knowledge, most Salesforce orgs use Communities to create a knowledge base. Creating a public knowledge base for Salesforce Knowledge in Salesforce Classic requires Sites and Visualforce.

Publishing Cycle

Salesforce Knowledge Articles move through a publishing cycle from their creation to their deletion. The publishing cycle includes three different statuses: `Draft` is the stage when a new article is being created or an existing one is being updated. Articles with the `Online` status are draft articles that have been published and are now available to their different channels. Eventually, when a published article is at the end of its life, it can be moved to the `Archived` status or sent back to `Draft` to be updated in a subsequent version.

Working with Articles in the API

Articles are available through the [KnowledgeArticleVersion](#) and [KnowledgeArticle](#) objects in the API. They both represent an article but provide different capabilities.

KnowledgeArticleVersion

Every new draft article in Salesforce Knowledge has a version number. When an article is published and you want to update it, you can create a new `Draft` with a distinct version number. Each version has its own ID. Once the updated version is ready to be published, it replaces the former one and updates the version number. You can access the content of an article version using the `KnowledgeArticleVersion` object and filter on its `Draft` or `Online` status. For example, the following query returns the title of the `Draft` version of all the articles across all article types in United States English:

```
SELECT Title
FROM KnowledgeArticleVersion
WHERE PublishStatus='Draft'
AND language = 'en_US'
```

Articles are also auto-assigned an Article Number, which is not a unique identifier to an individual article, but an identifier to a master article and all of its available translations.

 **Note:** Both the master version (the knowledge article with `IsMasterLanguage = 1`) and the translations are `KnowledgeArticleVersion` objects.

KnowledgeArticle

Unlike `KnowledgeArticleVersion`, the ID of a `KnowledgeArticle` record is identical irrespective of the article's version (status). Where the `KnowledgeArticleVersion` object provides API access to an article's custom field values, the `KnowledgeArticle` object provides API access to an article's metadata fields.

The article record is the parent container of all versions of an article, whatever the publishing status (draft, published, archived) and the language. While `KnowledgeArticle` and `KnowledgeArticleVersion` represent any article in the knowledge base, use the concrete representation of these objects for the specific articles. In Lightning Knowledge, these concrete representations default to `Knowledge__ka` (for the Knowledge article) and `Knowledge__kav` (for the Knowledge article version). In Salesforce Classic, use `<Article Type>__ka` and `<Article Type>__kav`.

The following Lightning Knowledge query returns the title for all the published FAQ articles in United States English:

```
SELECT Title
FROM Knowledge__kav
WHERE PublishStatus='online'
AND Language = 'en_US'
AND RecordTypeId = '<specify RecordTypeId for FAQ here>'
```

The following Salesforce Classic query returns the title for all the published offers in United States English:

```
SELECT Title
FROM FAQ__kav
WHERE PublishStatus='online'
AND language ='en_US'
```

Data Categories Overview

Data categories are organized by category group and let:

- Users classify and find records.
- Administrators control access to records.

Salesforce Knowledge uses data categories to classify articles and make them easier to find. For example, to classify articles by sales regions and products, create two category groups: Sales Regions, Products. The Sales Regions category group could consist of a geographical hierarchy, such as All Sales Regions as the top level and North America, Europe, and Asia at the second level. The Products group could have All Products as the top level and Phones, Computers, and Printers at the second.

Working with Data Categories in the API



The following table lists API resources for working with data categories.

Name	Type	Description
Knowledge__DataCategorySelection	Object	Gives access to article categorization in Lightning Knowledge.
ArticleType__DataCategorySelection	Object	Gives access to article categorization in Knowledge for Salesforce Classic.
QuestionDataCategorySelection	Object	Gives access to question categorization.
WITH DATA CATEGORY <i>filteringExpression</i>	SOQL clause	Filters articles depending on their status in the publishing cycle and their data categories. For more information, see the Salesforce SOQL and SOSL Reference Guide .
WITH DATA CATEGORY <i>DataCategorySpec</i>	SOSL clause	Finds articles based on their categorization. For more information, see the Salesforce SOQL and SOSL Reference Guide .
describeDataCategoryGroups()	Call	Retrieves available category groups for objects specified in the request.
describeDataCategoryGroupStructures()	Call	Retrieves available category groups along with their data category structure for objects specified in the request.
<code>describeDataCategoryGroups</code>	Apex method	Returns a list of the category groups associated with the specified objects. See the Apex Developer Guide .
<code>describeDataCategoryGroupStructures</code>	Apex method	Returns available category groups along with their data category structure for objects specified in the request. See the Apex Developer Guide .

API End-of-Life

Salesforce is committed to supporting each API version for a minimum of three years from the date of first release. In order to mature and improve the quality and performance of the API, versions that are more than three years old might cease to be supported.

When an API version is to be deprecated, advance notice is given at least one year before support ends. Salesforce will directly notify customers using API versions planned for deprecation.

-  **Note:** Version 20.0 of REST API and SOAP API have now been deprecated and are no longer supported. You can continue to access this legacy API version until Summer '22 is released, at which point this legacy version will be retired and will become unavailable. For more information, see this Knowledge Article: [Salesforce Platform API Versions 7.0 through 20.0 Retirement](#).
-  **Note:** Versions 21.0 through 30.0 of REST API and SOAP API will be deprecated in the Summer '22 release. For more information, see this Knowledge Article: [Salesforce Platform API Versions 21.0 through 30.0 Retirement](#).

CHAPTER 2 Salesforce Knowledge SOAP API

In this chapter ...

- Knowledge SOAP API Objects
- Knowledge SOAP API Calls

You can manage articles and get support information with Salesforce Objects using the SOAP API.

Knowledge SOAP API Objects

SOAP API object resources for working with Salesforce Knowledge.

IN THIS SECTION:

[KnowledgeArticle](#)

Provides read-only access to an article and the ability to delete the master article. This object is available in API version 19.0 and later.

[KnowledgeArticleVersion](#)

Provides a global view of standard article fields across all types of articles depending on their version. This object is available in API version 18.0 and later.

[Knowledge__DataCategorySelection](#)

Represents a data category that classifies an article. This object is available in API version 39.0 and later.

[Knowledge__Feed](#)

Represents the feed for a knowledge article. This object is available in API version 39.0 and later.

[Knowledge__kav](#)

Provides access to the concrete object that represents a Knowledge article version. This object is available in API version 39.0 and later.

[Knowledge__ka](#)

Provides access to the concrete object that represents a Knowledge article, the parent object for article versions. This object is available in API version 39.0 and later.

[KnowledgeArticleVersionHistory](#)

Enables read-only access to the full history of an article. This object is available in API version 25.0 and later.

[KnowledgeArticleViewStat](#)

Provides statistics on the number of views for the specified article across all article types. This object is read-only and available in API version 20.0 and later.

[KnowledgeArticleVoteStat](#)

Provides the weighted rating for the specified article on a scale of 1 to 5 across all article types. This object is read-only and available in API version 20.0 and later.

[CaseArticle](#)

Represents the association between a Case and a KnowledgeArticle. This object is available in API version 20.0 and later.

[LinkedArticle](#)

Represents a knowledge article that is attached to a work order, work order line item, or work type. This object is available in API version 37.0 and later.

[RecentlyViewed](#)

Represents records or list views that the current user has recently viewed or referenced (by viewing a related record). List views are available in API version 29.0 and later.

[SearchPromotionRule](#)

Represents a promoted search term, which is one or more keywords that you associate with a Salesforce Knowledge article. When a user's search query includes these keywords, the associated article is returned first in search results. This object is available in API version 31.0 and later.

[TopicAssignment](#)

Represents the assignment of a topic to a specific feed item, record, or file. This object is available in API version 28.0 and later.

KnowledgeArticle

Provides read-only access to an article and the ability to delete the master article. This object is available in API version 19.0 and later.

Unlike KnowledgeArticleVersion, the ID of a KnowledgeArticle record is identical irrespective of the article's version (status).

[Knowledge__ka](#) on page 38 is derived from this object.

Supported Calls

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

Special Access Rules

Knowledge must be enabled in your org. A user must have the View Articles permission enabled. Salesforce Knowledge users, unlike customer and partner users, must also be granted the `Knowledge User` feature license.

Fields

Field Name	Details
ArchivedById	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the user who archived the article.</p>
ArchivedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date when the article was archived.</p>
ArticleNumber	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p>Description The unique number automatically assigned to the article when it's created. You can't change the format or value for this field.</p>
CaseAssociationCount	<p>Type int</p>

Field Name	Details
	<p>Properties Filter, Group, Sort</p> <p>Description The number of cases attached to the article.</p>
FirstPublishedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date when the article was first published.</p>
LastPublishedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date when the article was last published.</p>
LastReferencedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The timestamp when the current user last accessed this record, a record related to this record, or a list view.</p>
LastViewedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The timestamp when the current user last viewed this record or list view. If this value is null, the user might have only accessed this record or list view (<code>LastReferencedDate</code>) but not viewed it.</p>
MasterLanguage	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p>

Field Name	Details
	<p>Description</p> <p>The article's original language. Only accessible if your knowledge base supports multiple languages.</p>
MigratedToFromArticle	<p>Type</p> <p>string</p> <p>Properties</p> <p>Filter, Group, Nillable, Sort</p> <p>Description</p> <p>The ID for the corresponding pre- or post-migration article. Contains values only in orgs that migrate from Knowledge in Salesforce Classic to Lightning Knowledge. This field is available in API version 45.0 and later.</p>
TotalViewCount	<p>Type</p> <p>int</p> <p>Properties</p> <p>Filter, Group, Nillable, Sort</p> <p>Description</p> <p>Total number of views for this article. This field is available in API version 39.0 and later.</p>

Usage

Use this object to query or retrieve articles. KnowledgeArticle can be used in a SOQL clause, but doesn't provide access to the fields from the article. Provides read-only access to an article and the ability to delete the master article.

Usage for SOQL with KnowledgeArticle

To expose the `migrated_to_from_id` column on **KnowledgeArticle** and **KnowledgeArticleVersion** to the sObject API: expose **MigratedToFromArticle** in **KnowledgeArticle**.

For SOQL:

- To filter by `MigratedToFromArticle`, remove any other filters.
- When filtering by `MigratedToFromArticle`, use the '=' or 'IN' operator.
- When filtering by `MigratedToFromArticle`, the value can't be null or empty.

KnowledgeArticleVersion

Provides a global view of standard article fields across all types of articles depending on their version. This object is available in API version 18.0 and later.

Use this object to:

- Query or search generically across multiple types of articles.

- Filter on a specific version.
- Update standard fields, those fields that are updateable, in draft versions.

When you query on the archived article, the results include both the article and the article's archived versions.

[Knowledge__kav](#) on page 30 is derived from this object.

Supported Calls

`describeLayout()`, `describeSObjects()`, `query()`, `retrieve()`, `search()`

Note:

- You can only update draft versions.
- You can't update draft translations with the `knowledgeManagement` REST API.
- For Lightning Knowledge, to create, update, or delete a Knowledge article version, use the call on `Knowledge__kav`. For example, to delete, use `Knowledge__kav.delete()`.
- For Knowledge in Salesforce Classic, to create, update, or delete a Knowledge article version, use the call on `ArticleType__kav`, where `ArticleType` is the name of the article's type. For example, to delete, use `ArticleType__kav.delete()`.

Special Access Rules

Knowledge must be enabled in your org. A user must have the View Articles permission enabled. Salesforce Knowledge users, unlike customer and partner users, must also be granted the `Knowledge User` feature license.

Fields

Field Name	Details
ArchivedById	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the user who archived the article.</p>
ArchivedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date the article version was archived.</p>
ArticleArchivedById	<p>Type reference</p>

Field Name	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the user who archived the article.</p>
ArticleArchivedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date the article was archived.</p>
ArticleCaseAttachCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of cases where this article is attached.</p>
ArticleCreatedById	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the user who created the article.</p>
ArticleCreatedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date the article was created.</p>
ArticleMasterLanguage	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The article's original language. Only accessible if your knowledge base supports multiple languages.</p>

Field Name	Details
ArticleNumber	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, Sort</p> <p>Description The unique number automatically assigned to the article when it's created. You can't change the format or value for this field.</p>
ArticleTotalViewCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Total number of views for the article.</p>
ArticleType	<p>Type string</p> <p>Properties Defaulted on create, Filter</p> <p>Description Indicates the API Name of the article type. The <code>ArticleType</code> is assigned to the article when it's created. You can't change the value of this field. This field is available in orgs using Knowledge in Salesforce Classic in API version 26.0 and later.</p>
AssignedById	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the user who assigned the article.</p>
AssignedToId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the user assigned to the article.</p>
AssignmentDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p>

Field Name	Details
	<p>Description The date the article was assigned to a user.</p>
AssignmentDueDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The due date when an article is assigned.</p>
AssignmentNote	<p>Type textarea</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Notes to the assignee from the user who assigned the article.</p>
FirstPublishedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date when the article was first published.</p>
IsLatestVersion	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the article is the most current version. (<code>true</code>) or not (<code>false</code>). This field can be <code>true</code> on the online or published version, a draft version in the master language, a draft version in a translation, and the latest archived version. However, you can't filter by (<code>PublishState='Online'</code>) and (<code>IsLatestVersion=false</code>) because the online version is also the latest version. This field is available in API version 24.0 and later.</p>
IsMasterLanguage	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p>

Field Name	Details
	<p>Description</p> <p>Indicates whether the article has one or more translations associated with it (<code>true</code>) or not (<code>false</code>). Only accessible if your knowledge base supports multiple languages.</p>
IsOutOfDate	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Defaulted on create, Filter, Group, Sort</p> <p>Description</p> <p>Indicates whether the source article has been updated since this translated version was created (<code>true</code>) or not (<code>false</code>). Only accessible if your knowledge base supports multiple languages.</p>
IsVisibleInApp	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Defaulted on create, Filter, Group, Sort</p> <p>Description</p> <p>Required. Indicates whether the article is visible in the Articles tab (<code>true</code>) or not (<code>false</code>).</p>
IsVisibleInCsp	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Defaulted on create, Filter, Group, Sort</p> <p>Description</p> <p>Required. Indicates whether the article is visible in the Customer Portal (<code>true</code>) or not (<code>false</code>).</p>
IsVisibleInPkb	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Defaulted on create, Filter, Group, Sort</p> <p>Description</p> <p>Required. Indicates whether the article is visible in the public knowledge base (<code>true</code>) or not (<code>false</code>).</p>
IsVisibleInPrm	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Defaulted on create, Filter, Group, Sort</p>

Field Name	Details
	<p>Description Required. Indicates whether the article is visible in the partner portal (<code>true</code>) or not (<code>false</code>).</p>
KnowledgeArticleId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the article independent from its version. The value for this field is retrieved from the <code>Id</code> field of the KnowledgeArticle object.</p>
Language	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The language that the article is written in, such as <code>French</code> or <code>Chinese (Traditional)</code>. Querying or searching articles in SOSL require that you specify the <code>Language</code> field in the <code>WHERE</code> clause. The language must be the same for all article types. Before API version 47.0, you must include the <code>Language</code> field to filter queries on Knowledge article versions. In API version 47.0 and later, you can filter queries on Knowledge article versions with or without <code>Language</code> depending on what you are querying.</p>
LastPublishedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date when the article was last published.</p>
MasterVersionId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the source article, if the article is the translation of a source article. Only accessible if your knowledge base supports multiple languages.</p>

Field Name	Details
MigratedToFromArticleVersion	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID for the corresponding pre- or post-migration article version. Contains values only in orgs that migrate from Classic to Lightning Knowledge. Available in API version 43.0 and later.</p>
OwnerId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the article's owner.</p>
PublishStatus	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Restricted picklist, Sort</p> <p>Description The publication status for the article:</p> <ul style="list-style-type: none"> • Draft: any draft articles. • Online: articles published in Salesforce Knowledge. • Archived: archived articles. <p>A user must have the “Manage Articles” permission enabled to use Online.</p> <p>Article queries and searches in SOQL or SOSL require that you specify either the PublishStatus or the Id field in the WHERE clause. You can search for only one publication status per article type in a single SOSL query. When searching for articles with a PublishStatus of Archived, also check that IsLatestVersion equals false in your WHERE clause.</p>
SourceId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the source from which the article was created (Case or Reply).</p>
Summary	<p>Type textarea</p>

Field Name	Details
	<p>Properties Filter, Nillable, Sort</p> <p>Description Summary of the article. Maximum size is 1000 characters.</p>
Title	<p>Type string</p> <p>Properties Filter, Group, idLookup, Sort</p> <p>Description Required. Article's title. Maximum size is 255 characters.</p>
TranslationCompletedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Date and time when the article was last translated. Only accessible if your knowledge base supports multiple languages.</p>
TranslationExportedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Date and time when the article was last exported for translation. Only accessible if your knowledge base supports multiple languages.</p>
TranslationImportedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Date and time when the article was last imported for translation. Only accessible if your knowledge base supports multiple languages.</p>
UrlName	<p>Type string</p> <p>Properties Filter, Group, idLookup, Sort</p>

Field Name	Details
	<p>Description</p> <p>Required. Represents the article's URL. Can contain alphanumeric characters and hyphens but can't begin or end with a hyphen. This value should be unique regardless of context. (For example, a unique value allows you to get expected results when running an Apex test with <code>SeeAllData</code> set to <code>false</code>.) <code>UrlName</code> is case-sensitive and its maximum size is 255 characters.</p>
<code>ValidationStatus</code>	<p>Type</p> <p>picklist</p> <p>Properties</p> <p>Defaulted on create, Filter, Group</p> <p>Description</p> <p>Shows whether the content of the article has been validated. Possible values are <code>Validated</code> and <code>Not Validated</code>. The default value is <code>Not Validated</code>. This field is available in API version 24.0 or later.</p>
<code>VersionNumber</code>	<p>Type</p> <p>int</p> <p>Properties</p> <p>Group, Sort</p> <p>Description</p> <p>The number assigned to a version of an article. This field is available in API version 24.0 and later.</p>

Usage

Use this object to query, retrieve, or search for articles across all types of articles depending on their version. You can update draft master articles. Also, you can delete articles that aren't drafts. Client applications can use `KnowledgeArticleVersion` with `describeDataCategoryGroups()` and `describeDataCategoryGroupStructures()` to return the category groups and the category structure associated with Salesforce Knowledge.

To access an article independent of its version, use the `KnowledgeArticle` object.

In Lightning Knowledge, the type of article is determined by the `RecordType` field on the concrete derived object (for example, [Knowledge__kav](#) on page 30). For Knowledge in Salesforce Classic, the type of article is determined by the `ArticleType` field and the concrete derived object uses the prefix of the article type name (for example, `FAQ__kav` for the FAQ article type).

SOQL Samples

The following SOQL clause uses `KnowledgeArticleVersion` to query all published articles from all articles complying with the classification specified in the `WITH DATA CATEGORY` clause:

```
SELECT Title, Summary
FROM KnowledgeArticleVersion
WHERE PublishStatus='Online'
```

```
AND Language = 'en_US'
WITH DATA CATEGORY Geography__c ABOVE_OR_BELOW europe__c AND Product__c BELOW All__c
```

The following SOQL clause for Lightning Knowledge uses the `Offer` record type to limit the query to all draft articles:

```
SELECT Id, Title
FROM Knowledge__kav
WHERE PublishStatus='Draft'
AND Language = 'en_US'
AND RecordTypeId = '<specify RecordTypeId for Offer here>'
WITH DATA CATEGORY Geography__c AT (france__c,usa__c) AND Product__c ABOVE dsl__c
```

The following SOQL clause for Salesforce Classic uses the `Offer` article type to limit the query to all draft articles:

```
SELECT Id, Title
FROM Offer__kav
WHERE PublishStatus='Draft'
AND Language = 'en_US'
WITH DATA CATEGORY Geography__c AT (france__c,usa__c) AND Product__c ABOVE dsl__c
```

The following SOQL clause uses `KnowledgeArticleVersion` to query the IDs of all archived versions of a particular article:

```
SELECT Id
FROM KnowledgeArticleVersion
WHERE PublishStatus='Archived'
AND IsLatestVersion=false
AND KnowledgeArticleId='kA1D00000001PQ6KAM'
```

SOQL and SOSL with KnowledgeArticleVersion

- Filter on a single value of `PublishStatus` for best results. To find all versions of each article, omit the `PublishStatus` filter, but do filter on one or more primary key IDs. To retrieve all archived versions for a given article, specify a SOQL filter where `IsLatestVersion` is `false`.
- In API version 46.0 and earlier, queries without a filter on `PublishStatus` return published articles by default. In API version 47.0 and later, draft, published, and archived articles are returned when Lightning Knowledge is enabled.
- To support security, only users with the “View Draft Articles” permission see articles whose `PublishStatus` value is `Draft`. Similarly, only users with the “View Archived Articles” permission see articles whose `PublishStatus` value is `Archived`.
- Archived article versions are stored in the **Knowledge__kav** object. To query archived article versions, specify the article `Id` and set `IsLatestVersion='0'`.
- You can't use binding variables in Apex SOQL statements with `KnowledgeArticleVersion` objects. For example, the following SOQL statement causes a compilation error.

```
final String PUBLISH_STATUS_ONLINE = 'Online';
List<Knowledge__kav> articles = [
SELECT Id FROM Knowledge__kav
WHERE PublishStatus = :PUBLISH_STATUS_ONLINE
];
```

Instead, use dynamic SOQL as follows. See [Dynamic SOQL](#) in *Apex Developer Guide*.

```
final String PUBLISH_STATUS_ONLINE = 'Online';
final String q = 'SELECT Id, PublishStatus FROM Knowledge__kav
WHERE PublishStatus = :PUBLISH_STATUS_ONLINE';
List<Knowledge__kav> articles = Database.query(q);
```

Other Usage for SOQL and SOSL with KnowledgeArticleVersion

To expose the *migrated_to_from_id* on **KnowledgeArticle** and **KnowledgeArticleVersion** to the sObject API: expose **MigratedToFromArticleVersion** in **KnowledgeArticleVersion**.

- For SOQL:
 - To filter by **MigratedToFromArticleVersion**, remove any other filters.
 - When filtering by **MigratedToFromArticleVersion**, use the '=' or 'IN' operator.
 - When filtering by **MigratedToFromArticleVersion**, the value can't be null or empty.
- SOSL doesn't support **MigratedToFromArticleVersion**.

Associated Objects


This object has the following associated objects. Unless noted, they are available in the same API version as this object.

KnowledgeArticleVersionHistory

History is available for tracked fields of the object.

Knowledge__DataCategorySelection

Represents a data category that classifies an article. This object is available in API version 39.0 and later.

 **Note:** By default, the prefix for this object name is `Knowledge` and that is the value shown in this reference. However, this prefix can be modified by changing the **Object Name** for the `Knowledge__kav` object in Object Manager.

Supported Calls

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

Special Access Rules

Lightning Knowledge must be enabled in your org.


Fields

Field	Details
DataCategoryGroupName	<p>Type datacategorygroupreference</p> <p>Properties Create</p> <p>Description Unique name of the data category group which has categories associated with the article.</p>
DataCategoryName	<p>Type datacategorygroupreference</p>

Field	Details
	<p>Properties Create</p> <p>Description Unique name of the data category associated with the article.</p>
ParentId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the article associated with the data category selection.</p>

Usage

Every article in Salesforce Knowledge can be categorized. A data category selection represents a category that has been selected to classify an article. You can use this object to query and manage article categorization in your organization. Client applications can create a categorization for an article with a Draft status. They can also delete and query article categorizations.

 **Note:** When using this object to classify an article, you can't select both a category (for example USA) and one of its descendants (California) or ascendant categories (North America). In this case, only the first category is selected.

Knowledge__Feed

Represents the feed for a knowledge article. This object is available in API version 39.0 and later.

For additional information about feeds, see FeedItem.

 **Note:** By default, the prefix for this object name is Knowledge and that is the value shown in this reference. However, this prefix can be modified by changing the **Object Name** for the Knowledge__kav object in Object Manager.

Supported Calls



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

Special Access Rules

Lightning Knowledge must be enabled in your org.

Fields

Field	Details
BestCommentId	<p>Type reference</p>

Field	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the comment marked as best answer on a question post.</p>
Body	<p>Type textarea</p> <p>Properties Nillable, Sort</p> <p>Description The body of the feed item. Required when <code>Type</code> is <code>TextPost</code> or <code>AdvancedTextPost</code>. Optional when <code>Type</code> is <code>ContentPost</code> or <code>LinkPost</code>.</p> <p>Although a value for <code>Body</code> is not required for the <code>ContentPost</code> type, an attachment is required. If an attachment isn't present, the type changes to <code>TextPost</code> or <code>AdvancedTextPost</code>, depending on the API version. <code>TextPost</code> and <code>AdvancedTextPost</code> do require a value for <code>Body</code>.</p> <p> Tip: See the <code>IsRichText</code> field for a list of HTML tags supported in the body of rich text posts.</p>
CommentCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of comments associated with this feed item.</p> <p> Tip: In a feed that supports pre-moderation, <code>CommentCount</code> isn't updated until a comment is published. For example, say that you comment on a post that already has one published comment and your comment triggers moderation. Now there are two comments on the post, but the count says there's only one. In a moderated feed, comments aren't counted until approved by an admin or someone with <code>Can Approve Feed Post and Comment</code> or <code>Modify All Data</code>.</p> <p>Feed moderation has implications on how you retrieve feed comments. In a moderated feed, rather than retrieving comments by looping through <code>CommentCount</code>, go through pagination until the end of comments is returned.</p>
InsertedById	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p>

Field	Details
LinkUrl	<p>Type url</p> <p>Properties Nillable, Sort</p> <p>Description The URL of a <code>LinkPost</code>.</p>
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description ID of the Knowledge article to which the feed item is related.</p>
RelatedRecordId	<p>Type reference</p> <p>Properties Group, Nillable, Sort</p> <p>Description ID of the <code>ContentVersion</code> record associated with a <code>ContentPost</code>. For WDC thanks posts, it's the ID of the <code>WorkThanks</code> object associated with a <code>RypplePost</code>. This field is typically null for all posts except <code>ContentPost</code> and <code>RypplePost</code>.</p> <p>For example, set this field to an existing <code>ContentVersion</code> ID and post it to a feed with <code>Type</code> set to <code>ContentPost</code>.</p>
Title	<p>Type string</p> <p>Properties Group, Nillable, Sort</p> <p>Description The title of the feed item. When the <code>Type</code> is <code>LinkPost</code>, the <code>LinkUrl</code> is the URL and this field is the link name. The <code>Title</code> field can be updated on posts of <code>Type</code> <code>QuestionPost</code>.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description The type of feed item. Except for <code>ContentPost</code>, <code>LinkPost</code>, and <code>TextPost</code>, don't create feed items of other types directly from the API.</p>


Field**Details**

- **ActivityEvent**—indirectly generated event when a user or the API adds a Task associated with a feed-enabled parent record (excluding email tasks on cases). Also occurs when a user or the API adds or updates a Task or Event associated with a case record (excluding email and call logging).

For a recurring Task with CaseFeed disabled, one event is generated for the series only. For a recurring Task with CaseFeed enabled, events are generated for the series and each occurrence.
- **AdvancedTextPost**—created when a user posts a group announcement and, in Lightning Experience as of API version 39.0 and later, when a user shares a post.
- **AnnouncementPost**—Not used.
- **ApprovalPost**—generated when a user submits an approval.
- **BasicTemplateFeedItem**—Not used.
- **CanvasPost**—a post made by a canvas app posted on a feed.
- **CollaborationGroupCreated**—generated when a user creates a public group.
- **CollaborationGroupUnarchived**—Not used.
- **ContentPost**—a post with an attached file.
- **CreatedRecordEvent**—generated when a user creates a record from the publisher.
- **DashboardComponentAlert**—generated when a dashboard metric or gauge exceeds a user-defined threshold.
- **DashboardComponentSnapshot**—created when a user posts a dashboard snapshot on a feed.
- **LinkPost**—a post with an attached URL.
- **PollPost**—a poll posted on a feed.
- **ProfileSkillPost**—generated when a skill is added to a user’s Chatter profile.
- **QuestionPost**—generated when a user posts a question.
- **ReplyPost**—generated when Chatter Answers posts a reply.
- **RypplePost**—generated when a user creates a Thanks badge in WDC.
- **TextPost**—a direct text entry on a feed.
- **TrackedChange**—a change or group of changes to a tracked field.
- **UserStatus**—automatically generated when a user adds a post. Deprecated.


The following values appear in the `Type` picklist for all feed objects but apply only to CaseFeed:

- **AttachArticleEvent**—generated event when a user attaches an article to a case.
- **CallLogPost**—generated event when a user logs a call for a case through the user interface. CTI calls also generate this event.
- **CaseCommentPost**—generated event when a user adds a case comment for a case object.
- **ChangeStatusPost**—generated event when a user changes the status of a case.
- **ChatTranscriptPost**—generated event when Chat transcript is saved to a case.

Field	Details
	<ul style="list-style-type: none"> • <code>EmailMessageEvent</code>—generated event when an email related to a case object is sent or received. • <code>FacebookPost</code>—generated when a Facebook post is created from a case. Deprecated. • <code>MilestoneEvent</code>—generated when a case milestone is completed or reaches violation status. • <code>SocialPost</code>—generated when a social post is created from a case. <p> Note: If you set <code>Type</code> to <code>ContentPost</code>, also specify <code>ContentData</code> and <code>ContentFileName</code>.</p>

Knowledge__kav

Provides access to the concrete object that represents a Knowledge article version. This object is available in API version 39.0 and later.

 **Note:** By default, the prefix for this object name is `Knowledge` and that is the value shown in this reference. However, this prefix can be modified by changing the **Object Name** for the `Knowledge__kav` object in Object Manager.

This object is derived from [KnowledgeArticleVersion](#) on page 13.

Supported Calls

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

Special Access Rules

Lightning Knowledge must be enabled in your org. A user must have the View Articles permission enabled. Salesforce Knowledge users, unlike customer and partner users, must also be granted the `Knowledge User` feature license.

Fields

Field	Details
<code>ArchivedById</code>	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the user who archived the article.</p>
<code>ArchivedDate</code>	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p>

Field	Details
	<p>Description The date the article version was archived.</p>
ArticleArchivedById	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the user who archived the article.</p>
ArticleArchivedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date the article was archived.</p>
ArticleCaseAttachCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The number of cases where this article is attached.</p>
ArticleCreatedById	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the user who created the article.</p>
ArticleCreatedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date the article was created.</p>
ArticleMasterLanguage	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p>

Field	Details
	<p>Description The article's original language. Only accessible if your knowledge base supports multiple languages.</p>
ArticleNumber	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, Sort</p> <p>Description The unique number automatically assigned to the article when it's created. You can't change the format or value for this field.</p>
ArticleTotalViewCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Total number of views for the article.</p>
AssignedById	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the user who assigned the article.</p>
AssignedToId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the user assigned to the article.</p>
AssignmentDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date the article was assigned to a user.</p>
AssignmentDueDate	<p>Type dateTime</p>

Field	Details
	<p>Properties Filter, Nillable, Sort</p> <p>Description The due date when an article is assigned.</p>
AssignmentNote	<p>Type textarea</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Notes to the assignee from the user who assigned the article.</p>
FirstPublishedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date when the article was first published.</p>
IsLatestVersion	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the article is the most current version. (<code>true</code>) or not (<code>false</code>). This field can be <code>true</code> on the online or published version, a draft version in the master language, a draft version in a translation, and the latest archived version. However, you can't filter by (<code>PublishState='Online'</code>) and (<code>IsLatestVersion=false</code>) because the online version is also the latest version. This field is available in API version 24.0 and later.</p>
IsMasterLanguage	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the article has one or more translations associated with it (<code>true</code>) or not (<code>false</code>).</p>
IsOutOfDate	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p>

Field	Details
	<p>Description</p> <p>Indicates whether the source article has been updated since this translated version was created (<code>true</code>) or not (<code>false</code>).</p>
IsVisibleInApp	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Defaulted on create, Filter, Group, Sort</p> <p>Description</p> <p>Required. Indicates whether the article is visible in the Articles tab (<code>true</code>) or not (<code>false</code>).</p>
IsVisibleInCsp	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description</p> <p>Required. Indicates whether the article is visible in the Customer Portal (<code>true</code>) or not (<code>false</code>).</p>
IsVisibleInPkb	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description</p> <p>Required. Indicates whether the article is visible in the public knowledge base (<code>true</code>) or not (<code>false</code>).</p>
IsVisibleInPrm	<p>Type</p> <p>boolean</p> <p>Properties</p> <p>Create, Defaulted on create, Filter, Group, Sort, Update</p> <p>Description</p> <p>Required. Indicates whether the article is visible in the partner portal (<code>true</code>) or not (<code>false</code>).</p>
KnowledgeArticleId	<p>Type</p> <p>reference</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>The ID of the article independent from its version. The value for this field is retrieved from the <code>Id</code> field of the KnowledgeArticle object.</p>

Field	Details
Language	<p>Type picklist</p> <p>Properties Create, Filter, Group, Restricted picklist, Sort</p> <p>Description The language that the article is written in, such as <code>French</code> or <code>Chinese (Traditional)</code>. Querying or searching articles in SOSL require that you specify the <code>Language</code> field in the <code>WHERE</code> clause. The language must be the same for all article types. Before API version 47.0, you must include the <code>Language</code> field to filter queries on Knowledge article versions. In API version 47.0 and later, you can filter queries on Knowledge article versions with or without <code>Language</code> depending on what you are querying.</p>
LastPublishedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date when the article was last published.</p>
MasterVersionId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the source article, if the article is the translation of a source article. Only accessible if your knowledge base supports multiple languages.</p>
MigratedToFromArticleVersion	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID for the corresponding pre- or post-migration article version. Contains values only in orgs that migrate from Classic to Lightning Knowledge. Available in API version 43.0 and later.</p>
OwnerId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p>


Field	Details
	<p>Description The ID of the article's owner.</p>
PublishStatus	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group, Restricted picklist, Sort</p> <p>Description The publication status for the article:</p> <ul style="list-style-type: none"> • Draft: any draft articles. • Online: articles published in Salesforce Knowledge. • Archived: archived articles. <p>A user must have the “Manage Articles” permission enabled to use Online.</p> <p>Article queries and searches in SOQL or SOSL require that you specify either the PublishStatus or the Id field in the WHERE clause. You can search for only one publication status per article type in a single SOSL query. When searching for articles with a PublishStatus of Archived, also check that IsLatestVersion equals false in your WHERE clause.</p>
RecordTypeId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description Indicates the API Name that describes the type of article. Use the record type to determine the article structure and other settings for different types of content.</p>
SourceId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the source from which the article was created (Case or Reply).</p>
Summary	<p>Type textarea</p> <p>Properties Create, Filter, Nillable, Sort, Update</p> <p>Description Summary of the article. Maximum size is 1000 characters.</p>

Field	Details
Title	<p>Type string</p> <p>Properties Create, Defaulted on create, Filter, Group, idLookup, Sort, Update</p> <p>Description Required. Article's title. Maximum size is 255 characters.</p>
TranslationCompletedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Date and time when the article was last translated. Only accessible if your knowledge base supports multiple languages.</p>
TranslationExportedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Date and time when the article was last exported for translation. Only accessible if your knowledge base supports multiple languages.</p>
TranslationImportedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Date and time when the article was last imported for translation. Only accessible if your knowledge base supports multiple languages.</p>
UrlName	<p>Type string</p> <p>Properties Create, Filter, Group, idLookup, Sort, Update</p> <p>Description Required. Represents the article's URL. Can contain alphanumeric characters and hyphens but can't begin or end with a hyphen. This value should be unique regardless of context. (For example, a unique value allows you to get expected results when running an Apex test with <code>SeeAllData</code> set to <code>false</code>.) <code>UrlName</code> is case-sensitive and its maximum size is 255 characters.</p>

Field	Details
ValidationStatus	<p>Type picklist</p> <p>Properties Defaulted on create, Filter, Group</p> <p>Description Shows whether the content of the article has been validated. Possible values are <code>Validated</code> and <code>Not Validated</code>. The default value is <code>Not Validated</code>. This field is available in API version 24.0 or later.</p>
VersionNumber	<p>Type int</p> <p>Properties Group, Sort</p> <p>Description The number assigned to a version of an article. This field is available in API version 24.0 and later.</p>

Knowledge__ka

Provides access to the concrete object that represents a Knowledge article, the parent object for article versions. This object is available in API version 39.0 and later.

 **Note:** By default, the prefix for this object name is `Knowledge` and that is the value shown in this reference. However, this prefix can be modified by changing the **Object Name** for the `Knowledge__kav` object in Object Manager.

This object is derived from [KnowledgeArticle](#) on page 11.

Supported Calls

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

Special Access Rules

Lightning Knowledge must be enabled in your org. A user must have the View Articles permission enabled. Salesforce Knowledge users, unlike customer and partner users, must also be granted the `Knowledge User` feature license.

Fields

Field	Details
ArchivedById	<p>Type reference</p>

Field	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID of the user who archived the article.</p>
ArchivedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date when the article was archived.</p>
ArticleNumber	<p>Type string</p> <p>Properties Autonumber, Defaulted on create, Filter, idLookup, Sort</p> <p>Description The unique number automatically assigned to the article when it's created. You can't change the format or value for this field.</p>
CaseAssociationCount	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number of cases attached to the article.</p>
FirstPublishedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date when the article was first published.</p>
LastPublishedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The date when the article was last published.</p>

Field	Details
LastReferencedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The timestamp when the current user last accessed this record, a record related to this record, or a list view.</p>
LastViewedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort</p> <p>Description The timestamp when the current user last viewed this record or list view. If this value is null, the user might have only accessed this record or list view (LastReferencedDate) but not viewed it.</p>
MasterLanguage	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The article's original language. Only accessible if your knowledge base supports multiple languages.</p>
MigratedToFromArticle	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID for the corresponding pre- or post-migration article. Contains values only in orgs that migrate from Knowledge in Salesforce Classic to Lightning Knowledge. This field is available in API version 45.0 and later.</p>
TotalViewCount	<p>Type int</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description Total number of views for this article. This field is available in API version 39.0 and later.</p>

KnowledgeArticleVersionHistory

Enables read-only access to the full history of an article. This object is available in API version 25.0 and later.

Knowledge__VersionHistory is derived from this object. To access this derived object, [turn on field history tracking](#) for Knowledge objects.

Supported Calls

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

You can also enable `delete()` in API version 42.0 and later. See [Enable delete of Field History and Field History Archive](#).

Special Access Rules

Knowledge must be enabled in your org. This object respects field, entity, and record-level security. You must have at least “Read” permission on the article type or the field to access its history. For data category security, Salesforce determines access based on the categorization of the online version of an article. If there is no online version, then security is applied based on the archived version, followed by the security of the draft version.

Fields

Field Name	Details
<code>DataType</code>	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The type of data that is tracked in the history table. This field is available in API version 50.0 and later.</p>
<code>EventType</code>	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The type of event that is tracked in the history table.</p>
<code>FieldName</code>	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted picklist, Sort</p> <p>Description Name of the tracked field.</p>

Field Name	Details
Language	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The language that the article is written in, such as <code>French</code> or <code>Chinese (Traditional)</code>. Querying or searching articles in SOSL requires that you specify the <code>Language</code> field in the WHERE clause. The language must be the same for all article types.</p>
NewValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The new value of the field that was changed.</p>
OldValue	<p>Type anyType</p> <p>Properties Nillable, Sort</p> <p>Description The most recent value of the field before it was changed.</p>
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The ID of the article.</p>
ParentObjectType	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The type of object that contains the field.</p>
VersionId	<p>Type reference</p>

Field Name	Details
	<p>Properties Filter, Group, Nillable, Sort</p> <p>Description The ID assigned to a version of the article.</p>
VersionNumber	<p>Type int</p> <p>Properties Filter, Group, Sort</p> <p>Description The number assigned to a version of an article. This field is available in API version 24.0 and later.</p>

Usage

Use this object to query events in the history of an article. For example, you can retrieve the number of edits a particular user has made to an article, how many times the article has been published, and so on.

KnowledgeArticleViewStat

Provides statistics on the number of views for the specified article across all article types. This object is read-only and available in API version 20.0 and later.

Knowledge__ViewStat is derived from this object.

Supported Calls

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

Special Access Rules

Knowledge must be enabled in your org. Users must have access to the published version of an article to retrieve its views. For more information on published article version, see the `PublishStatus` field in `KnowledgeArticleVersion`.

Fields

Field Name	Details
Channel	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p>

Field Name	Details
	<p>Description</p> <p>The channel where the article is viewed:</p> <ul style="list-style-type: none"> • <code>AllChannels</code> for article views across all channels. • <code>App</code> for the internal Salesforce Knowledge application. • <code>Pkb</code> for article views in public knowledge base. • <code>Csp</code> for Customer Portal. • <code>Prm</code> for article view in partner portal.
<code>NormalizedScore</code>	<p>Type</p> <p>double</p> <p>Properties</p> <p>Filter, Nillable, Sort</p> <p>Description</p> <p>Article's weighted views in the selected channel. The article with most views has a score of 100. Other article views are then calculated relative to this highest view score. For example, if the best read article has 2000 views and another has 1000. The first one gets a score of 100 while the second gets 50.</p>
<code>ParentId</code>	<p>Type</p> <p>reference</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>ID of the viewed article. This corresponds to a KnowledgeArticle record.</p>
<code>ViewCount</code>	<p>Type</p> <p>int</p> <p>Properties</p> <p>Filter, Group, Sort</p> <p>Description</p> <p>The number of unique views an article has received in the selected channel. An article with a high number of views may not always have a high normalized score. The normalized score for an article is calculated based on views over time, with more recent views earning a higher score. This field is available in API version 27.0 and later.</p>

Usage

Use this object to query or retrieve statistics for article views.

Alternatively, client applications can use the article type `API Name` followed by `__ViewStat` to query or retrieve most viewed articles from a specific article type.

SOQL Samples

The following SOQL clause uses KnowledgeArticleViewStat to query all the article views in Salesforce Knowledge and return the related articles:

```
SELECT Id, NormalizedScore, Parent.Id
      FROM KnowledgeArticleViewStat where Channel = 'App'
      ORDER BY NormalizedScore
```

Use the following clause to restrict your query to Offer articles for the Offer article type:

```
SELECT Id, NormalizedScore, Parent.Id
      FROM Offer__ViewStat where Channel = 'App'
      ORDER BY NormalizedScore
```

KnowledgeArticleVoteStat

Provides the weighted rating for the specified article on a scale of 1 to 5 across all article types. This object is read-only and available in API version 20.0 and later.

Knowledge__VoteStat is derived from this object.

Supported Calls

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

Special Access Rules

Knowledge must be enabled in your org. Users must have access to the published version of an article to retrieve its votes. For more information on published article version, see the `PublishStatus` field in KnowledgeArticleVersion

Fields

Field Name	Details
Channel	<p>Type picklist</p> <p>Properties Filter, Group, Restricted picklist, Sort</p> <p>Description The channel where the article is rated:</p> <ul style="list-style-type: none"> • <code>AllChannels</code> for article views across all channels. • <code>App</code> for the internal Salesforce Knowledge application. • <code>Pkb</code> for article views in public knowledge base. • <code>Csp</code> for Customer Portal. • <code>Ppm</code> for article view in partner portal.

Field Name	Details
NormalizedScore	<p>Type double</p> <p>Properties Filter, Nillable, Sort</p> <p>Description Article's weighted score on a scale of 1 to 5. A higher score means more votes. Articles without recent votes trend towards an average rating of three stars.</p>
ParentId	<p>Type reference</p> <p>Properties Filter, Group, Sort</p> <p>Description The rated article. This corresponds to a KnowledgeArticle record.</p>

Usage

Use this object to query or retrieve the rating for an article.

Alternatively, client applications can use the article type `API Name` followed by `__VoteStat` to query or retrieve the rating for an article for a specific article type.

SOQL Samples

See KnowledgeArticleViewStat.

CaseArticle

Represents the association between a Case and a KnowledgeArticle. This object is available in API version 20.0 and later.

Supported Calls

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

Special Access Rules

Access to this object is controlled by the parent Case and KnowledgeArticle. However, when querying, access is only controlled by the parent Case.

Customer Portal users can't access this object.

Fields

Field	Details
ArticleLanguage	<p>Type picklist</p> <p>Properties Filter, Restricted picklist</p> <p>Description The language of the article associated with the case.</p>
ArticleVersionNumber	<p>Type int</p> <p>Properties Create, Group, Nillable</p> <p>Description The number assigned to a version of an article. This field is available in API version 24.0 and later.</p>
CaseId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the Case associated with the KnowledgeArticle.</p>
IsSharedByEmail	<p>Type int</p> <p>Properties Create, Group, Nillable</p> <p>Description Indicates that the article has been shared with the customer through an email.</p>
KnowledgeArticleId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description ID of the KnowledgeArticle associated with the Case.</p>

Usage

This object represents the association of a knowledge article with a Case. An article is associated with a case when it's relevant to a specific issue, when it helps an agent solve the case, or when the agent sends the article to a customer.

You can use this object to include case-article associations in Apex and Visualforce.

You can't update this object via the API. If you attempt to create a record that matches an existing record, the create request simply returns the existing record.

LinkedArticle

Represents a knowledge article that is attached to a work order, work order line item, or work type. This object is available in API version 37.0 and later.

Supported Calls

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

Special Access Rules

Knowledge must be enabled in your org. Field Service must be enabled. Only users that have access to the Knowledge article and the parent record linked to it can access this object.

In Knowledge in Salesforce Classic, only Field Service objects such as Work Order, Work Type, and Work Order Line Item are supported for linked articles. In Lightning Knowledge, other social objects such as Chat, Messaging, Voice Call, and Social Post are supported for linked articles.

Fields

Field Name	Details
CurrencyIsoCode	<p>Type picklist</p> <p>Properties Create, Defaulted on create, Filter, Group, Nillable, Restricted picklist, Sort, Update</p> <p>Description Available only for orgs with the multicurrency feature enabled. Contains the ISO code for any currency allowed by the organization.</p>
KnowledgeArticleId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description The ID of the Knowledge article attached to the record. The label in the user interface is Knowledge Article ID.</p>

Field Name	Details
KnowledgeArticleVersionId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description The version of the Knowledge article attached to the record. This field lists the title of the attached version and links to the version. The label in the user interface is Article Version.</p> <p>When you attach an article to a work order, that version of the article stays associated with the work order, even if later versions are published. If needed, you can detach and reattach an article to a work order to link the latest version.</p>
LinkedEntityId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort</p> <p>Description The ID of the record that the Knowledge article is attached to. The label in the user interface is Linked Record ID.</p>
Name	<p>Type string</p> <p>Properties Create, Filter, Group, idLookup, Sort, Update</p> <p>Description The title of the article. The label in the user interface is Article Title.</p>
RecordTypeId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p> <p>Description The ID of the article's record type, if used. This field is only available for Lightning Knowledge.</p>
Type	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description (Read only) The type of record that the Knowledge article is attached to. For example, work order. The label in the user interface is Linked Object Type.</p>

Usage

Admins can customize linked articles' page layouts, fields, validation rules, and more from the Linked Articles page in Setup.

Associated Objects

This object has the following associated objects. Unless noted, they are available in the same API version as this object.

LinkedArticleFeed

Feed tracking is available for the object.

LinkedArticleHistory

History is available for tracked fields of the object.

RecentlyViewed

Represents records or list views that the current user has recently viewed or referenced (by viewing a related record). List views are available in API version 29.0 and later.


Supported Calls

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

Special Usage Rules

The RecentlyViewed object doesn't support the Event, Task, Report, KnowledgeArticle, and Article objects.

The RecentlyViewed object supports only certain objects, and supports list views only for those supported objects. Supported objects have the fields `LastReferencedDate` and `LastViewedDate`.

 **Note:** RecentlyViewed records for users who are members of several communities can't be retrieved automatically into a map via Apex. This is because records of a user with different networks can result in duplicate IDs that maps don't support.

Fields

Field	Details
Alias	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The alias on the record.</p>
Email	<p>Type email</p> <p>Properties Filter, Group, Nillable, Sort</p>

Field	Details
	<p>Description The email address on the record.</p>
FirstName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The first name on the record. If the recently viewed record is a user, the value is the user's first name.</p>
Id	<p>Type ID</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description The ID of the recently viewed record or list view.</p>
IsActive	<p>Type boolean</p> <p>Properties Defaulted on create, Filter, Group, Sort</p> <p>Description Indicates whether the recently viewed record is an active user (<code>true</code>) or not (<code>false</code>). This field contains a value only if the recently viewed record is a user.</p>
LastName	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The last name on the record.</p>
LastReferencedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp when the current user last accessed this record, a record related to this record, or a list view.</p>

Field	Details
LastViewedDate	<p>Type dateTime</p> <p>Properties Filter, Nillable, Sort, Update</p> <p>Description The timestamp when the current user last viewed this record or list view. If this value is null, the user might have only accessed this record or list view (<code>LastReferencedDate</code>) but not viewed it.</p>
Name	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The name on the recently viewed record or list view. If the recently viewed record is a user, contact, or lead, the value is a concatenation of the <code>firstname</code> and <code>lastname</code> field values.</p>
NetworkId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description ID of the Experience Cloud site that this group is part of. This field is available only if digital experiences is enabled in your org.</p> <p>You can add a <code>NetworkId</code> only when creating a group. You can't change or add a <code>NetworkId</code> for an existing group. This field is available in API version 27.0 and later.</p>
Phone	<p>Type phone</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description The phone number on the record.</p>
ProfileId	<p>Type reference</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description If the recently viewed record is a user, this value is the user's profile ID.</p> <p>This is a relationship field.</p>

Field	Details
	<p>Relationship Name Profile</p> <p>Relationship Type Lookup</p> <p>Refers To Profile</p>
Title	<p>Type string</p> <p>Properties Filter, Group, Nillable, Sort</p> <p>Description If the recently viewed record is a user, this value is the title of the user; for example CFO or CEO.</p>
Type	<p>Type picklist</p> <p>Properties Filter, Group, Nillable, Restricted, picklistSort</p> <p>Description The object type for this recently viewed record or list view. Valid values include any standard or custom objects that RecentlyViewed supports.</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. This is a relationship field.</p> <p>Relationship Name UserRole</p> <p>Relationship Type Lookup</p> <p>Refers To UserRole</p>

Usage

This object provides a heterogeneous list of different object types. The list consists of recently viewed records, records that were recently referenced (a related record was viewed), or recently viewed list views. A record is considered viewed when the user sees the record

details, but not when the user sees the record 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, use this object 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 and 200 list views per object. `RecentlyViewed` data is retained for 90 days, after which it's removed on a periodic basis.

Use this query in your code to retrieve a list of all the records and list views 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 ('Account', 'Contact', 'Plan__c')
ORDER BY LastViewedDate DESC
```

This query retrieves a list of all recently viewed contacts with contact-specific fields, such as the contact's account name, and the custom website field. Records are ordered from most to least recent.

```
SELECT Account.Name, Title, Email, Phone, Website__c
FROM Contact
WHERE LastViewedDate != NULL
ORDER BY LastViewedDate DESC
```

SearchPromotionRule

Represents a promoted search term, which is one or more keywords that you associate with a Salesforce Knowledge article. When a user's search query includes these keywords, the associated article is returned first in search results. This object is available in API version 31.0 and later.

Supported Calls

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

Special Access Rules

A user must have the "Manage Promoted Search Terms" permission.

Fields

Field Name	Details
PromotedEntityId	<p>Type reference</p> <p>Properties Create, Filter, Group, Nillable, Sort, Update</p>

Field Name	Details
	<p>Description</p> <p>The ID of the KnowledgeArticleVersion that the promoted search term is associated with. The article must be in published status.</p>
Query	<p>Type</p> <p>string</p> <p>Properties</p> <p>Create, Filter, Group, Sort, Update</p> <p>Description</p> <p>The text of the promoted search term. Maximum length: 100 characters.</p> <p>You can associate the same promoted search term with multiple articles. If the user's search matches the promoted term, all associated articles are promoted in search results, ordered by relevancy. For best results, create promoted search terms selectively and limit the number of articles that are promoted per term.</p>

Usage

Use this object to optimize article search results in Salesforce Knowledge.

TopicAssignment

Represents the assignment of a topic to a specific feed item, record, or file. This object is available in API version 28.0 and later.

Administrators must enable topics for objects before users can add topics to records of that object type. Topics for most objects are available in API version 30.0 and later. Topics for ContentDocument are available in API version 37.0 and later.

Supported Calls

`create()`, `describeSObjects()`, `delete()`, `getDeleted()`, `getUpdate()`, `query()`, `retrieve()`

Fields

Field Name	Details
EntityId	<p>Type</p> <p>reference</p> <p>Properties</p> <p>Create, Filter, Group, Sort</p> <p>Description</p> <p>Identifier of the feed item, record, or file.</p> <p>This is a polymorphic relationship field.</p> <p>Relationship Name</p> <p>Entity</p>

Field Name**Details****Relationship Type**

Lookup

Refers To

Account, Asset, Campaign, Case, Contact, ContentDocument, Contract, Event, FeedItem, Lead, Opportunity, Order, ProductItem, ProductItemTransaction, ProductRequest, ProductRequestLineItem, ProductRequired, ProductTransfer, ResourceAbsence, ResourcePreference, ReturnOrder, ReturnOrderLineItem, ServiceAppointment, ServiceResource, ServiceResourceSkill, ServiceTerritory, ServiceTerritoryMember, Shift, Shipment, Solution, Task, WorkOrder, WorkOrderLineItem

EntityKeyPrefix

Type

string

Properties

Filter, Group, idLookup, Sort

Description

The first three digits of the `EntityID` field, which identify the object type (account, opportunity, etc). This read-only field is available in API version 32.0 and later.

Interface label is "Record Key Prefix," which appears only in reports.

EntityType

Type

string

Properties

Filter, Group, Nillable, Sort

Description

The standard name for the object type (account, opportunity, etc). This read-only field is available in API version 33.0 and later.



Note: Querying topic assignments for the ManagedContentVersion entity type isn't supported.

Interface label is "Object Type," which appears only in reports.



Tip: In most cases, you should use this field rather than `EntityKeyPrefix`, which exists primarily to support older reports.

NetworkId

Type

reference

Properties

Create, Filter, Group, Nillable, Sort

Description

Identifier of the community to which the TopicAssignment belongs. This field is available only if digital experiences is enabled in your org.

Field Name	Details
TopicId	<p>Type reference</p> <p>Properties Create, Filter, Group, Sort</p> <p>Description Identifier of the topic. This is a relationship field.</p> <p>Relationship Name Topic</p> <p>Relationship Type Lookup</p> <p>Refers To Topic</p>

Usage

Use this object to query the assignments of topics to feed items, records, or files. To assign or remove topics, you must have the “Assign Topics” permission.

In SOQL `SELECT` syntax, this object supports nested semi-joins, allowing queries on Knowledge articles assigned to specific topics. For example:

```
SELECT parentId FROM KnowledgeArticleViewStat
  WHERE parentId in (SELECT KnowledgeArticleId FROM KnowledgeArticleVersion
  WHERE publishStatus = 'Online' AND language = 'en_US'
  AND Id in (select EntityId from TopicAssignment where TopicId ='0T0xx0000000xxx'))
```

No SOQL limit if logged-in user has “View All Data” permission. If not, do one of the following:

- Specify a `LIMIT` clause of 1,100 records or fewer.
- Filter on `Id` or `Entity` when using a `WHERE` clause with “=”.

Knowledge SOAP API Calls

SOAP API calls for working with Salesforce Knowledge.

IN THIS SECTION:

[describeKnowledge\(\)](#)

Retrieves the Knowledge language settings in the organization.

[describeDataCategoryGroups\(\)](#)

Retrieves available category groups for objects specified in the request.

[describeDataCategoryGroupStructures\(\)](#)

Retrieves available category groups along with their data category structure for objects specified in the request.

[search\(\)](#)

Executes a text search in your organization's data.

describeKnowledge ()

Retrieves the Knowledge language settings in the organization.

Syntax

```
KnowledgeSettings result = _connection.describeKnowledgeSettings();
```

Usage

Use this call to describe the existing Knowledge language settings, including the default Knowledge language, supported languages, and a list of Knowledge language information. You can also use KnowledgeSettings in the Metadata API to obtain similar information.

Sample Code—Java

This sample shows how to retrieve the Knowledge language settings. It returns the default Knowledge language, a list of Knowledge supported language, including the language code and whether it's an active Knowledge language.

```
public void describeKnowledgeSettingsSample() {
    try {

        // Make the describe call for KnowledgeSettings
        KnowledgeSettings result = connection.describeKnowledgeSettings();

        // Get the properties of KnowledgeSettings
        System.out.println("Knowledge default language: " + result.getDefaultLanguage());
        for (KnowledgeLanguageItem lang : result.getLanguages()) {
            System.out.println("Language: " + lang.getName());
            System.out.println("Active: " + lang.isActive());
        }
    } catch (ConnectionException ex) {
        ex.printStackTrace();
    }
}
```

Sample Code—C#

This sample shows how to retrieve the Knowledge language settings. It returns the default Knowledge language, a list of Knowledge supported language, including the language code and whether it's an active Knowledge language.

```
public void describeKnowledgeSettingsSample() {
    try {

        // Make the describe call for KnowledgeSettings
        KnowledgeSettings result = connection.describeKnowledgeSettings();

        // Get the properties of KnowledgeSettings
```



```

Console.WriteLine("Knowledge default language: " + result.getDefaultLanguage());
for (KnowledgeLanguageItem lang : result.getLanguages()) {
Console.WriteLine("Language: " + lang.getName());
Console.WriteLine("Active: " + lang.isActive());
}
} catch (SoapException ex) {
ex.printStackTrace();
}
}

```

Response

KnowledgeSettings

describeDataCategoryGroups ()

Retrieves available category groups for objects specified in the request.

Syntax

```
DescribeDataCategoryGroupResult[] = connection.describeDataCategoryGroups() (string[]
sObjectTypes);
```

Usage

Use this call to describe the available category groups for the objects specified in the request. This call can be used with the `describeDataCategoryGroupStructures ()` call to describe all the categories available for a specific object. For additional information about data categories, see “Work with Data Categories” in the Salesforce online help.

Sample Code—Java

This sample shows how to retrieve the data category groups associated with:

- Salesforce Knowledge articles
- Questions from the Answers feature

It returns the name, label and description of a category group and the name of the associated `sObject` (article or question). It also returns the number of data categories in the data category group.

```

public void describeDataCategoryGroupsSample() {
    try {
        // Make the describe call for data category groups
        DescribeDataCategoryGroupResult[] results =
            connection.describeDataCategoryGroups(new String[] {
                "KnowledgeArticleVersion", "Question"});

        // Get the properties of each data category group
        for (int i = 0; i < results.length; i++) {
            System.out.println("sObject: " +
                results[i].getSObject());
            System.out.println("Group name: " +

```

```

results[i].getName());
System.out.println("Group label: " +
results[i].getLabel());
System.out.println("Group description: " +
(results[i].getDescription()==null? "" :
results[i].getDescription()));
System.out.println("Number of categories: " +
results[i].getCategoryCount());
}
} catch (ConnectionException ce) {
ce.printStackTrace();
}
}
}

```

Sample Code—C#

This sample shows how to retrieve the data category groups associated with:

- Salesforce Knowledge articles
- Questions from the Answers feature

It returns the name, label and description of a category group and the name of the associated `subject` (article or question). It also returns the number of data categories in the data category group.

```

public void describeDataCategoryGroups() {
    try {
        // Make the describe call for data category groups
        DescribeDataCategoryGroupResult[] results =
binding.describeDataCategoryGroups(new String[] {
"KnowledgeArticleVersion", "Question"});

        // Get the properties of each data category group
        for (int i = 0; i < results.Length; i++) {
            Console.WriteLine("sObject: " +
results[i].subject);
            Console.WriteLine("Group name: " +
results[i].name);
            Console.WriteLine("Group label: " +
results[i].label);
            Console.WriteLine("Group description: " +
(results[i].description==null? "" :
results[i].description));
            Console.WriteLine("Number of categories: " +
results[i].categoryCount);
        }
    } catch (SoapException e) {
        Console.WriteLine("An unexpected error has occurred: " +
e.Message + "\n" + e.StackTrace);
    }
}

```

Arguments

Name	Type	Description
sObjectTypes	string[]	<p>The specified value can be:</p> <ul style="list-style-type: none"> • <code>KnowledgeArticleVersion</code>—to retrieve category groups associated with article types. • <code>Question</code>—to retrieve category groups associated with questions. <p>For additional information about articles and questions, see "Work with Articles and Translations" in the Salesforce online help.</p>

Response

DescribeDataCategoryGroupResult

Faults

InvalidSObjectFault

UnexpectedErrorFault

IN THIS SECTION:

[DescribeDataCategoryGroupResult](#)

The `describeDataCategoryGroups()` call returns a `DescribeDataCategoryGroupResult` object containing the list of the category groups associated with the specified objects.

DescribeDataCategoryGroupResult

The `describeDataCategoryGroups()` call returns a `DescribeDataCategoryGroupResult` object containing the list of the category groups associated with the specified objects.

Name	Type	Description
categoryCount	int	The number of visible data categories in the data category group.
description	string	The description of the data category group.
label	string	Label for the data category group in the Salesforce user interface.
name	string	The unique name used for API access to the data category group .
subject	string	The object associated with the data category group.

describeDataCategoryGroupStructures ()

Retrieves available category groups along with their data category structure for objects specified in the request.

Syntax

```
describeDataCategoryGroupStructures() [] = connection.
    describeDataCategoryGroupStructures() (DataCategoryGroupObjectTypePair[]
    pairs, boolean topCategoriesOnly)
```

Usage

Use this call to return the visible data category structure for the given object category group pairs. First use [describeDataCategoryGroups \(\)](#) to find the available category groups for the objects specified. From the returned list, choose the object category group pairs to pass as the input in `describeDataCategoryGroupStructures ()`. This call returns all the visible categories and data category structure as output. For additional information about data categories and data category visibility, see “Work with Data Categories” and “Data Category Visibility” in the Salesforce online help.

Sample Code—Java

This sample shows how to use sObject and data category group pairs to retrieve data categories for each pair. It calls `describeDataCategoryGroupStructures ()` with two pairs, KnowledgeArticleVersion/Regions and Question/Regions, and iterates through the results of this call. It gets the top categories for each result, which is “All”, and then gets the first-level child categories. The sample requires that you set up a data category group called *Regions* with some child categories and associate it with a knowledge article and questions. Alternatively, you can replace the data category group name in the sample if you want to use an existing data category group in your org that has a different name.

```
public void describeDataCategoryGroupStructuresSample() {
    try {
        // Create the data category pairs
        DataCategoryGroupObjectTypePair pair1 =
        new DataCategoryGroupObjectTypePair();
        DataCategoryGroupObjectTypePair pair2 =
        new DataCategoryGroupObjectTypePair();
        pair1.setSubject("KnowledgeArticleVersion");
        pair1.setDataCategoryGroupName("Regions");
        pair2.setSubject("Question");
        pair2.setDataCategoryGroupName("Regions");

        DataCategoryGroupObjectTypePair[] pairs =
        new DataCategoryGroupObjectTypePair[] {
            pair1,
            pair2
        };

        // Get the list of top level categories using the describe call
        DescribeDataCategoryGroupStructureResult[] results =
        connection.describeDataCategoryGroupStructures(
            pairs,
            false
        );

        // Iterate through each result and get some properties
        // including top categories and child categories
```

```

for (int i = 0; i < results.length; i++) {
DescribeDataCategoryGroupStructureResult result =
results[i];
String sObject = result.getSObject();
System.out.println("sObject: " + sObject);
System.out.println("Group name: " + result.getName());
System.out.println("Group label: " + result.getLabel());
System.out.println("Group description: " +
result.getDescription());

// Get the top-level categories
DataCategory[] topCategories = result.getTopCategories();

// Iterate through the top level categories and retrieve
// some information
for (int j = 0; j < topCategories.length; j++) {
DataCategory topCategory = topCategories[j];
System.out.println("Category name: " +
topCategory.getName());
System.out.println("Category label: " +
topCategory.getLabel());
DataCategory [] childCategories =
topCategory.getChildCategories();
System.out.println("Child categories: ");
for (int k = 0; k < childCategories.length; k++) {
System.out.println("\t" + k + ". Category name: " +
childCategories[k].getName());
System.out.println("\t" + k + ". Category label: " +
childCategories[k].getLabel());
}
}
} catch (ConnectionException ce) {
ce.printStackTrace();
}
}

```

Sample Code—C#

This sample shows how to use `sObject` and data category group pairs to retrieve data categories for each pair. It calls `describeDataCategoryGroupStructures()` with two pairs, `KnowledgeArticleVersion/Regions` and `Question/Regions`, and iterates through the results of this call. It gets the top categories for each result, which is "All", and then gets the first-level child categories. The sample requires that you set up a data category group called *Regions* with some child categories and associate it with a knowledge article and questions. Alternatively, you can replace the data category group name in the sample if you want to use an existing data category group in your org that has a different name.

```

public void describeDataCateogryGroupStructuresSample() {
    try {
        // Create the data category pairs
        DataCategoryGroupSubjectTypePair pair1 =
        new DataCategoryGroupSubjectTypePair();
        DataCategoryGroupSubjectTypePair pair2 =
        new DataCategoryGroupSubjectTypePair();

```

```

pair1.subject = "KnowledgeArticleVersion";
//pair1.setDataCategoryGroupName("Regions");
pair1.dataCategoryGroupName = "KBArticleCategories";
pair2.subject = "Question";
//pair2.setDataCategoryGroupName("Regions");
pair2.dataCategoryGroupName = "KBArticleCategories";

DataCategoryGroupSubjectTypePair[] pairs =
new DataCategoryGroupSubjectTypePair[] {
pair1,
pair2
};

// Get the list of top level categories using the describe call
DescribeDataCategoryGroupStructureResult[] results =
binding.describeDataCategoryGroupStructures(
pairs,
false
);

// Iterate through each result and get some properties
// including top categories and child categories
for (int i = 0; i < results.Length; i++) {
DescribeDataCategoryGroupStructureResult result =
results[i];
String sObject = result.subject;
Console.WriteLine("sObject: " + sObject);
Console.WriteLine("Group name: " + result.name);
Console.WriteLine("Group label: " + result.label);
Console.WriteLine("Group description: " +
result.description);

// Get the top-level categories
DataCategory[] topCategories = result.topCategories;

// Iterate through the top level categories and retrieve
// some information
for (int j = 0; j < topCategories.Length; j++) {
DataCategory topCategory = topCategories[j];
Console.WriteLine("Category name: " +
topCategory.name);
Console.WriteLine("Category label: " +
topCategory.label);
DataCategory [] childCategories =
topCategory.childCategories;
Console.WriteLine("Child categories: ");
for (int k = 0; k < childCategories.Length; k++) {
Console.WriteLine("\t" + k + ". Category name: " +
childCategories[k].name);
Console.WriteLine("\t" + k + ". Category label: " +
childCategories[k].label);
}
}
}
}

```

```

    }
    catch (SoapException e)
    {
        Console.WriteLine("An unexpected error has occurred: " +
            e.Message + "\n" + e.StackTrace);
    }
}

```

Arguments

Name	Type	Description
pairs	DataCategoryGroupObjectTypePair[]	Specifies a category group and an object to query. Visible data categories are retrieved for that object.
topCategoriesOnly	boolean	Indicates whether the call returns only the top (<code>true</code>) or all the categories (<code>false</code>) visible depending on the user's data category group visibility settings. For more information on data category group visibility, see Data Category Visibility in the Salesforce online help.

[DataCategoryGroupObjectTypePair](#) contains the following fields:

Name	Type	Description
dataCategoryGroupName	string	The unique name used for API access to the data category group.
subject	string	The object associated with the data category group

Response

[describeDataCategoryGroupStructures\(\)](#) []

Faults

[InvalidSObjectFault](#)

[UnexpectedErrorFault](#)

IN THIS SECTION:

[describeDataCategoryGroupStructures\(\)](#)

The `describeDataCategoryGroupStructures()` call returns an array of `DescribeDataCategoryGroupStructureResult` objects containing the category groups and categories associated with the specified objects.

describeDataCategoryGroupStructures ()

The `describeDataCategoryGroupStructures()` call returns an array of `DescribeDataCategoryGroupStructureResult` objects containing the category groups and categories associated with the specified objects.

Name	Type	Description
description	string	The description of the data category group.
label	string	The label for the data category group in the Salesforce user interface.
name	string	The unique name used for API access to the data category group.
subject	string	The object associated with the data category group.
topCategories	DataCategory []	A list of top level categories visible depending on the user's data category group visibility settings. For more information on data category group visibility, see "Data Category Visibility" in the Salesforce online help.

DataCategory

Name	Type	Description
childDataCategories	DataCategory []	A recursive list of visible sub categories in the data category.
label	string	The label for the data category in the Salesforce user interface.
name	string	The unique name used for API access to the data category.

search ()

Executes a text search in your organization's data.

Syntax

```
SearchResult = connection.search(String searchString);
```

Usage

Use [search \(\)](#) to search for records based on a search string. The search call supports searching custom objects. For an extensive discussion about the syntax and rules used for text searches, see the [Salesforce SOQL and SOSL Reference Guide](#).

Certain objects cannot be searched via the API, such as [Attachment](#) objects. To search an object via the `search()` call, the object must be configured as searchable (`isSearchable` is `true`). To determine whether an object can be searched, your client application can invoke the `describeObjects ()` call on the object and inspect its `searchable` property.

Sample Code—Java

This sample makes the `search ()` call by passing it a SOSL query, which returns contacts, leads, and accounts whose phone fields contain a specified value. Next, it gets the `sObject` records from the results and stores the records in arrays depending on the record type. Finally, it writes the fields of the returned contacts, leads, and accounts to the console.

```
public void searchSample() {
    try {
        // Perform the search using the SOSL query.
        SearchResult sr = connection.search(
```



```

"FIND {4159017000} IN Phone FIELDS RETURNING "
+ "Contact(Id, Phone, FirstName, LastName), "
+ "Lead(Id, Phone, FirstName, LastName), "
+ "Account(Id, Phone, Name)");

// Get the records from the search results.
SearchRecord[] records = sr.getSearchRecords();

ArrayList<Contact> contacts = new ArrayList<Contact>();
ArrayList<Lead> leads = new ArrayList<Lead>();
ArrayList<Account> accounts = new ArrayList<Account>();

// For each record returned, find out if it's a
// contact, lead, or account and add it to the
// appropriate array, then write the records
// to the console.
if (records.length > 0) {
for (int i = 0; i < records.length; i++) {
SObject record = records[i].getRecord();
if (record instanceof Contact) {
contacts.add((Contact) record);
} else if (record instanceof Lead) {
leads.add((Lead) record);
} else if (record instanceof Account) {
accounts.add((Account) record);
}
}
}

System.out.println("Found " + contacts.size() + " contacts.");
for (Contact c : contacts) {
System.out.println(c.getId() + ", " + c.getFirstName() + ", "
+ c.getLastName() + ", " + c.getPhone());
}
System.out.println("Found " + leads.size() + " leads.");
for (Lead d : leads) {
System.out.println(d.getId() + ", " + d.getFirstName() + ", "
+ d.getLastName() + ", " + d.getPhone());
}
System.out.println("Found " + accounts.size() + " accounts.");
for (Account a : accounts) {
System.out.println(a.getId() + ", " + a.getName() + ", "
+ a.getPhone());
}
} else {
System.out.println("No records were found for the search.");
}
} catch (Exception ce) {
ce.printStackTrace();
}
}

```

Sample Code—C#

This sample makes the `search()` call by passing it a SOSL query, which returns contacts, leads, and accounts whose phone fields contain a specified value. Next, it gets the sObject records from the results and stores the records in arrays depending on the record type. Finally, it writes the fields of the returned contacts, leads, and accounts to the console.

```
public void searchSample()
{
    try
    {
        // Perform the search using the SOSL query.
        SearchResult sr = binding.search(
            "FIND {4159017000} IN Phone FIELDS RETURNING "
            + "Contact(Id, Phone, FirstName, LastName), "
            + "Lead(Id, Phone, FirstName, LastName), "
            + "Account(Id, Phone, Name)");

        // Get the records from the search results.
        SearchRecord[] records = sr.searchRecords;

        List<Contact> contacts = new List<Contact>();
        List<Lead> leads = new List<Lead>();
        List<Account> accounts = new List<Account>();

        // For each record returned, find out if it's a
        // contact, lead, or account and add it to the
        // appropriate array, then write the records
        // to the console.
        if (records.Length > 0)
        {
            for (int i = 0; i < records.Length; i++)
            {
                sObject record = records[i].record;
                if (record is Contact)
                {
                    contacts.Add((Contact)record);
                }
                else if (record is Lead)
                {
                    leads.Add((Lead)record);
                }
                else if (record is Account)
                {
                    accounts.Add((Account)record);
                }
            }

            Console.WriteLine("Found " + contacts.Count + " contacts.");
            foreach (Contact c in contacts)
            {
                Console.WriteLine(c.Id + ", " +
                    c.FirstName + ", " +
                    c.LastName + ", " +
                    c.Phone);
            }
        }
    }
}
```

```

Console.WriteLine("Found " + leads.Count + " leads.");
foreach (Lead d in leads)
{
Console.WriteLine(d.Id + ", " +
d.FirstName + ", " +
d.LastName + ", " +
d.Phone);
}
Console.WriteLine("Found " + accounts.Count + " accounts.");
foreach (Account a in accounts)
{
Console.WriteLine(a.Id + ", " +
a.Name + ", " +
a.Phone);
}
}
else
{
Console.WriteLine("No records were found for the search.");
}
}
catch (SoapException e)
{
Console.WriteLine("An unexpected error has occurred: " +
e.Message + "\n" + e.StackTrace);
}
}

```

Arguments

Name	Type	Description
search	string	Search string that specifies the text expression to search for, the scope of fields to search, the list of objects and fields to retrieve, and the maximum number of records to return. For more information, see the Salesforce SOQL and SOSL Reference Guide .

Response

[SearchResult](#)

Fault

InvalidFieldFault

InvalidSObjectFault

MalformedSearchFault

UnexpectedErrorFault

IN THIS SECTION:

[SearchResult](#)

The `search()` call returns a `SearchResult` object, which has the following properties.

SearchResult

The `search()` call returns a `SearchResult` object, which has the following properties.

Name	Type	Description
<code>queryId</code>	string	Unique identifier for the <code>SOSL</code> search.
<code>searchRecords</code>	SearchRecord[]	Array of <code>SearchRecord</code> objects, each of which contains an <code>sObject</code> .
<code>searchResultsMetadata</code>	SearchResultsMetadata	Metadata for <code>SearchRecords</code> .

SearchRecord

Represents an individual record returned from a search.

Name	Type	Description
<code>record</code>	<code>sObject</code>	The individual record returned by the search.
<code>searchRecordMetadata</code>	SearchRecordMetadata	Metadata for <code>searchRecords</code> .
<code>snippet</code>	SearchSnippet	On the search results page, shows terms that match the search string, highlighted within the surrounding text.

SearchRecordMetadata

Metadata for search results at the record level.

Name	Type	Description
<code>searchPromoted</code>	boolean	Indicates that an article has been promoted in search results. Admins define promoted search terms by adding promoted terms to knowledge articles. Users who search for these keywords see the article first in search results. Available in API version 42.0 and later.
<code>spellCorrected</code>	boolean	Indicates that a record matches a spell-corrected search term. Appears in the response only when true.

SearchSnippet

Excerpts shown on search results pages for article, case, feed, and idea searches.

Name	Type	Description
<code>text</code>	string	The excerpt that contains the match for the search term.

Name	Type	Description
wholeFields	WholeFields	The list of highlighted fields.

WholeFields

Contains the complete text of each field that contains highlighting for terms that match the search query. The highlighted terms are surrounded by <mark> tags.

Name	Type	Description
name	string	The name of the highlighted field.
value	string	The highlighted text.

SearchResultsMetadata

Global metadata for the search result.

Name	Type	Description
entityMetadata	EntitySearchMetadata	Search results metadata at the object level.

EntitySearch/Metadata

Metadata for search results at the object level.

Name	Type	Description
fieldMetadata	FieldLevelSearchMetadata	Metadata for search results at the field level.
searchPromotedMetadata	EntitySearchPromotionMetadata	Metadata for search term promotion at the object level. Available in API version 42.0 and later.
spellCorrectionMetadata	EntitySpellCorrectionMetadata	Metadata for spelling correction at the object level.
entityName	string	Identifies the object.

FieldLevelSearchMetadata

Metadata for search results at the field level.

Name	Type	Description
name	string	The field name.
label	string	The field label.
type	string	The field type.

EntitySearchPromotionMetadata

Metadata for search term promotion at the object level. Appears in the response only when at least one article for an object is a promoted result. Available in API version 42.0 and later.

Name	Type	Description
<code>promotedResultCount</code>	int	Count of promoted article results at the object level.

EntitySpellCorrectionMetadata

Metadata for spelling correction at the object level. Appears in the response only when at least one record for an object matches a spell-corrected search term.

Name	Type	Description
<code>correctedQuery</code>	string	The spell-corrected search term.
<code>hasNonCorrectedResults</code>	boolean	If <code>true</code> , indicates that the user has access to at least one record that matches a search term that wasn't spell-corrected. Each object sometimes returns a different value.

CHAPTER 3 Salesforce Knowledge REST APIs

In this chapter ...

- [Invocable Actions for Lightning Knowledge](#)
- [Manage Knowledge with REST APIs](#)
- [Support Knowledge with REST API](#)

You can manage articles and get support information with REST APIs.

REST API uses the same underlying [data model](#) and [standard objects](#) as those in SOAP API. For generic REST API information see the [REST API Developer Guide](#).

Invocable Actions for Lightning Knowledge

Actions for Lightning Knowledge can be invoked from a REST endpoint. Use these actions to manage your articles and article versions.

To learn more about how to use invocable actions, see the [Actions Developer Guide](#).

IN THIS SECTION:

[Knowledge Actions](#)

Manage your Knowledge articles using invocable actions.

Knowledge Actions

Manage your Knowledge articles using invocable actions.

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

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



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

Supported REST HTTP Methods

URIs

[Archive Knowledge articles:](#)

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

[Assign Knowledge articles:](#)

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

[Create draft from online Knowledge articles:](#)

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

[Delete Knowledge articles:](#)

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

[Publish Knowledge articles:](#)

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

[Restore Knowledge article version:](#)

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

[Retrieve Smart Link URL:](#)

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

[Submit Knowledge article for translation:](#)

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

Formats

JSON, XML

HTTP Methods

GET, HEAD, POST

AuthenticationAuthorization: Bearer *token***Other Information**[Error Response Types](#) on page 84

Archive Knowledge Articles

URI: /services/data/v~~xx.x~~/actions/standard/archiveKnowledgeArticles**Table 1: Inputs**

Input	Details
articleVersionIdList	<p>Type string</p> <p>Description Required. Comma-separated article version ID list.</p>

Sample Input

The following code sample archives two articles:

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

Sample Output

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

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

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

```
[ {
  "actionName" : "archiveKnowledgeArticles",
  "errors" : null,
  "isSuccess" : false,
  "outputValues" : {
```

```

    "ka0RM00000004Ve" : "You can't perform this action. Be sure the action is valid for
the current state of the article, and that you have permission to perform it.",
    "ka0RM00000003do" : "Success"
  }
} ]

```

Assign Knowledge Articles

URI: /services/data/v~~xx~~.x/actions/standard/assignKnowledgeArticles

Table 2: Inputs

Input	Details
articleVersionIdList	<p>Type string</p> <p>Description Required. Comma-separated article version ID list.</p>
assigneeId	<p>Type ID</p> <p>Description Required. ID of the assigned user.</p>
assignAction	<p>Type string</p> <p>Description Required. Assign action. Valid actions are:</p> <ul style="list-style-type: none"> • ASSIGN_DRAFT_MASTER • ASSIGN_DRAFT_TRANSLATION
dueDate	<p>Type string</p> <p>Description Optional. Assigned due date.</p>
instruction	<p>Type string</p> <p>Description Optional. Instructions for the assignee.</p>
sendEmailNotification	<p>Type boolean</p> <p>Description Optional. Indicates whether to send an email notification. Defaults to <code>false</code>.</p>

Sample Input

The following code sample assigns two articles for translation:

```
{
  "inputs" : [
    {
      "articleVersionIdList" : [ "ka0RM00000004VeYAI", "ka0RM00000003doYAA" ]
      "assigneeId" : "005RM000000AAAAA4",
      "assignAction" : "ASSIGN_DRAFT_TRANSLATION"
    }
  ]
}
```

Sample Output

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

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

Create Draft from Online Knowledge Article

URI: /services/data/v`xx.x`/actions/standard/createDraftFromOnlineKnowledgeArticle

Table 3: Inputs

Input	Details
action	<p>Type string</p> <p>Description Required. Edit action for master language or translation articles. Valid actions are:</p> <ul style="list-style-type: none"> • EDIT_AS_DRAFT_ARTICLE • EDIT_AS_DRAFT_TRANSLATION
unpublish	<p>Type boolean</p> <p>Description Required. Indicates whether to keep the article published (<code>false</code>) or archive the published article (<code>true</code>). Use <code>false</code> to keep the current article version online and create a new draft. Use <code>true</code> to archive the current online version, which removes it from the knowledge base, and creates a new draft.</p>

Input	Details
articleVersionId	<p>Type string</p> <p>Description Article version ID. Required to create a draft from an online (published) translation. Optional to create a draft from the online master article if the Article ID is provided.</p>
articleId	<p>Type string</p> <p>Description Article ID. Required when creating a draft from the online (published) master article if the Article Version ID isn't provided.</p>

Sample Input

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

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

Sample Output

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

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

Delete Knowledge Articles

URI: /services/data/v~~xx~~.x/actions/standard/deleteKnowledgeArticles

Table 4: Inputs

Input	Details
articleVersionIdList	<p>Type string</p>

Input**Details****Description**

Required. Comma-separated article version ID list.

Sample Input

The following code sample deletes two articles:

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

Sample Output

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

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

Publish Knowledge Articles

URI: /services/data/v~~xx.x~~/actions/standard/publishKnowledgeArticles

Table 5: Inputs

Input	Details
articleVersionIdList	<p>Type string</p> <p>Description Required. Comma-separated article version ID list.</p>
pubAction	<p>Type string</p> <p>Description Required. Publish action. Valid actions are:</p> <ul style="list-style-type: none"> • PUBLISH_ARTICLE (which replaces the latest version) • PUBLISH_ARTICLE_NEW_VERSION (which creates a new version) • SCHEDULE_ARTICLE_FOR_PUBLICATION

Input	Details
	<ul style="list-style-type: none"> PUBLISH_TRANSLATION
pubDate	<p>Type string</p> <p>Description Optional. Scheduled publish date.</p>

Sample Input

The following code sample publishes two articles:

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

Sample Output

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

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

Restore Knowledge Article Version

URI: /services/data/v~~xx.x~~/actions/standard/restoreKnowledgeArticleVersion

Table 6: Inputs

Input	Details
action	<p>Type string</p> <p>Description Required. The only valid action is: RESTORE_KNOWLEDGE_ARTICLE_VERSION</p>
articleId	<p>Type string</p>

Input**Details****Description**

Required. Article ID.

versionNumber

Type

integer

Description

Optional. Version number of the archived article version to restore. Default is the latest archived version.

Sample Input

The following code restores the latest archived version:

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

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

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

The following code restores two archived articles:

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

Sample Output

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

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

Retrieve Smart Link URL

URI: /services/data/v~~xx.x~~/actions/standard/getArticleSmartLinkUrl

Table 7: Inputs

Input	Details
articleVersionId	<p>Type string</p> <p>Description Required. The ID of the Knowledge article version.</p>

Sample Input

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

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

Sample Output

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

```
[
  {
    "actionName": "getArticleSmartLinkUrl",
    "errors": null,
    "isSuccess": true,
    "outputValues": {
      "articleSmartLinkUrl": "https://example.lightning.force.com/lightning/articles/Knowledge/Test-Redirection-1"
    }
  }
]
```


Submit Knowledge Article for Translation

URI: /services/data/v~~xx~~.x/actions/standard/submitKnowledgeArticleForTranslation

Table 8: Inputs

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

Table 9: Outputs

Output	Details
articleId	<p>Type ID</p> <p>Description Article ID.</p>
language	<p>Type string</p> <p>Description Language code for the translation.</p>

Sample Input

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

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

Sample Output

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

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

Error Response Types

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

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

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

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

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

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

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

Manage Knowledge with REST APIs

REST APIs provide programmatic access to many actions you can perform on your Knowledge base, an article, and its translations.

IN THIS SECTION:

[Archive the Master Version of an Article](#)

Archives the master version of an article. The actions are defined by the field change you request on the resource. To archive the master version, use `"publishStatus": "Archived"`. To schedule a date for archiving, use `"archiveScheduleDate": <date>`.

[Assign a Task Related to a Master Article](#)

Assigns a task to a user for a master article, including due date and instructions. The actions are defined by the field change you request on the resource.

[Assign a Task Related to a Translation](#)

Assigns a task to a user for a translated article, including due date and instructions. The actions are defined by the field change you request on the resource.

[Delete a Master Version of an Article](#)

Deletes the master version of an article.

[Delete a Translated Version of an Article](#)

Deletes a translated version of an article.

[Edit an Online Version of a Master Article](#)

Creates a draft copy of the online version of a master article. This does not unpublish the online version.

[Get a List of Salesforce Site and Experience Cloud Site URLs for an Online Article](#)

Returns a list of article URLs in Salesforce Sites, Experience Cloud sites, or both. Article URLs from Salesforce Sites are returned if the article is visible in a public knowledge base. Article URLs from Experience Cloud sites are returned if the article is visible to partners or customers. This API creates the URLs for the **Insert URL into Email** Lightning action.

[Publish the Master Version of an Article](#)

Publishes the master version of an article.

[Restore an Archived Version of an Article](#)

Restores an archived version of the article. If `versionNumber` isn't specified, restores the latest version of the archived article.

[Retrieve Article Metadata](#)

Retrieves the metadata of an article.

[Retrieve Knowledge Language Settings](#)

Returns the existing Knowledge language settings, including the default knowledge language and a list of supported Knowledge language information.

[Retrieve a Version of an Article](#)

Retrieves the version ID of an article.

[Search for Metadata Elements of a Master Version](#)

Searches for metadata elements of the online master version of an article.

[Search for Metadata Elements of a Translated Version](#)

Searches for metadata elements of a translated version of an article.

[Set a Translated Article Version to Complete](#)

Sets a translated article version to complete.

[Set a Translated Article Version to Incomplete](#)

Sets a translated article version to incomplete.

[Submit an Article for Translation](#)

Submits an article for translation for multiple languages and blocks translations for inactive languages.

[Unpublish the Master Version of an Article](#)

Unpublishes the online master version of an article when there isn't an existing draft article.

[Unpublish the Online Version of a Translated Article](#)

Unpublishes the online version of a translated article.

[Parameterized Search](#)

Executes a simple RESTful search using parameters instead of a SOSL clause. Indicate parameters in a URL in the GET method. Or, use POST for more complex JSON searches.

[Search](#)

Executes the specified SOSL search. The search string must be URL-encoded.

[Search Scope and Order](#)

Returns an ordered list of objects in the default global search scope of a logged-in user. Global search keeps track of which objects the user interacts with and how often, and arranges the search results accordingly. Objects used most frequently appear at the top of the list

[Search Result Layouts](#)

Returns search result layout information for the objects in the query string. For each object, this call returns the list of fields displayed on the search results page as columns, the number of rows displayed on the first page, and the label used on the search results page.

[Search for Records Suggested by Autocomplete and Instant Results](#)

Returns a list of suggested records whose names match the user's search string. The suggestions resource provides autocomplete results and instant results for users to navigate directly to likely relevant records, before performing a full search.

[Search Suggested Article Title Matches](#)

Returns a list of Salesforce Knowledge article titles that match the user's search query string. Provides a shortcut to navigate directly to likely relevant articles before the user performs a search.

[Search Suggested Queries](#)

Returns a list of suggested searches based on the user's query string text matching searches that other users have performed in Salesforce Knowledge. Provides a way to improve search effectiveness, before the user performs a search.

Archive the Master Version of an Article

Archives the master version of an article. The actions are defined by the field change you request on the resource. To archive the master version, use `"publishStatus": "Archived"`. To schedule a date for archiving, use `"archiveScheduleDate" : <date>`.

Syntax

URI

```
/services/data/v25.0/knowledgeManagement/articleVersions/masterVersions/<versionID>
```

Formats

JSON, XML

HTTP Method

PATCH

Authentication

Authorization: OAuth *access_token*

Parameters

Parameter	Description
publishStatus	Publishing status of the article. Use <code>archived</code> .
archiveScheduleDate	Date to archive the article.

Input:

Archive:

```
{
  "publishStatus": "Archived"
}
```

Schedule for archiving (using GMT date format):

```
{
  "archiveScheduleDate" : "2012-04-19T07:00:00.000+0000"
}
```

Output:

HTTP status code 204 is returned when an existing record is updated.

Assign a Task Related to a Master Article

Assigns a task to a user for a master article, including due date and instructions. The actions are defined by the field change you request on the resource.

Syntax

URI

```
/services/data/v25.0/knowledgeManagement/articleVersions/masterVersions/<versionID>
```

Formats

JSON, XML

HTTP Method

PATCH

Authentication

Authorization: OAuth *access_token*

Parameters

Parameter	Description
assigneeId	Assigns the master article to a user ID or a queue ID.
dueDate	Date that the task is due.
instruction	Instructions for the task.

Input:

```
{
  "assigneeId": "05Dxx0000dsads",
  "dueDate": "2012-04-19T07:00:00.000+0000",
  "instruction": "Please review."
}
```

Output:

HTTP status code 204 is returned when an existing record is updated.

Assign a Task Related to a Translation

Assigns a task to a user for a translated article, including due date and instructions. The actions are defined by the field change you request on the resource.

Syntax

URI

```
/services/data/v25.0/knowledgeManagement/articleVersions/translations/<translationVersionId>
```

Formats

JSON, XML

HTTP Method

PATCH

Authentication

Authorization: OAuth *access_token*

Parameters

Parameter	Description
assigneeId	Assigns the master article to a user ID or a queue ID.
dueDate	Date that the task is due.
instruction	Instructions for the task.

Input:

```
{
  "assigneeId": "05Dxx0000dsads",
  "dueDate": "2012-04-19T07:00:00.000+0000",
  "instruction": "Please review."
}
```

Output:

HTTP status code 204 is returned when an existing record is updated.

Delete a Master Version of an Article

Deletes the master version of an article.

Syntax

URI

```
/services/data/v25.0/knowledgeManagement/articleVersions/masterVersions/<versionID>
```

Formats

JSON, XML

HTTP Method

DELETE

Authentication

Authorization: OAuth *access_token*

Parameters

None

Input:

None required

Output:

HTTP status code 204 is returned when an existing record is deleted.

Delete a Translated Version of an Article

Deletes a translated version of an article.

Syntax

URI

```
/services/data/v25.0/knowledgeManagement/articleVersions/translations/<versionID>
```

Formats

JSON, XML

HTTP Method

DELETE

Authentication

Authorization: OAuth *access_token*

Parameters

None

Input

None needed

Output

HTTP status code 204 is returned when an existing record is deleted.

Edit an Online Version of a Master Article

Creates a draft copy of the online version of a master article. This does not unpublish the online version.

Syntax

URI

```
/services/data/v25.0/knowledgeManagement/articleVersions/masterVersions
```

Formats

JSON, XML

HTTP Method

POST

Authentication

Authorization: OAuth *access_token*

Parameters

Parameter	Description
articleId	The ID of the article.

Input:

```
{
  "articleId": <articleID>
}
```

Get a List of Salesforce Site and Experience Cloud Site URLs for an Online Article

Returns a list of article URLs in Salesforce Sites, Experience Cloud sites, or both. Article URLs from Salesforce Sites are returned if the article is visible in a public knowledge base. Article URLs from Experience Cloud sites are returned if the article is visible to partners or customers. This API creates the URLs for the **Insert URL into Email** Lightning action.

The org must have Lightning Knowledge enabled, and **Allow users to share articles via public URLs** enabled in Knowledge Settings. In addition, the API user must have read access on Knowledge.

Available in API versions 46.0 and later.

Syntax

URI

```
/services/data/v46.0/knowledgeManagement/articles/siteListForOnlineArticle?articleId=articleId&language=xx_XX
```

Formats

JSON, XML

HTTP Method

GET

Authentication

Authorization: OAuth *access_token*

Parameters

Parameter	Description
articleId	Either the 15- or 18-digit version of the article version ID.
language	Language for the selected article.

Input

None

Publish the Master Version of an Article

Publishes the master version of an article.

The publishing actions are defined by the field change you request on the resource. To publish a minor version, use "publishStatus": "Online". To publish a major version, use "publishStatus": "Online" and "versionNumber": "NextVersion". To schedule a publication date, use "publishScheduleDate" : <date>.

Syntax

URI

```
/services/data/v25.0/knowledgeManagement/articleVersions/masterVersions/<versionId>
```

Formats

JSON, XML

HTTP Method

PATCH

Authentication

Authorization: OAuth *access_token*

Parameters

Parameter	Description
<code>publishStatus</code>	Publishing status of the article. Use <code>online</code> .
<code>versionNumber</code>	Version of the article.
<code>publishScheduleDate</code>	Date to publish the article.

Input

Publish a minor version:

```
{
  "publishStatus": "Online"
}
```

Publish a major version:

```
{
  "publishStatus": "Online",
  "versionNumber": "NextVersion"
}
```

Schedule for publication (using GMT date format):


```
{
  "publishScheduleDate" : "2012-05-19T07:00:00.000+0000"
}
```

Output

HTTP status code 204 is returned when an existing record is updated.

Restore an Archived Version of an Article

Restores an archived version of the article. If `versionNumber` isn't specified, restores the latest version of the archived article.

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

Syntax

URI

```
/services/data/v25.0/knowledgeManagement/articleVersions/masterVersions/
```

Formats

JSON, XML

HTTP Method

POST

Authentication

Authorization: OAuth *access_token*

Parameters

Parameter	Description
<code>articleId</code>	The ID of the article.
<code>versionNumber</code>	Version of the article. If this field is not specified, the latest version of the archived article is restored.

Input:

```
{
  "articleId": "<articleID>",
  "versionNumber": <number>
}
```

Retrieve Article Metadata

Retrieves the metadata of an article.

Syntax

URI

```
/services/data/v25.0/knowledgeManagement/articles/<articleId>
```

Formats

JSON, XML

HTTP Method

GET

Authentication

Authorization: OAuth *access_token*

Parameters

None

Input

None required

Retrieve Knowledge Language Settings

Returns the existing Knowledge language settings, including the default knowledge language and a list of supported Knowledge language information.

Syntax

URI

```
/services/data/vXX.X/knowledgeManagement/settings
```

Available since release

31.0

Formats

JSON, XML

HTTP methods

GET

AuthenticationAuthorization: Bearer *token***Request body**

None required

Request parameters

None

Example for Getting KnowledgeSettings

```
curl https://MyDomainName.my.salesforce.com/services/data/v55.0/knowledgeManagement/settings  
-H "Authorization: Bearer token"
```

Example JSON Response Body

```
{  
  "defaultLanguage" : "en_US",  
  "knowledgeEnabled" : true,  
  "languages" : [ {  
    "active" : true,  
    "name" : "en_US"  
  }, {  
    "active" : true,  
    "name" : "it"  
  }, {  
    "active" : true,  
    "name" : "zh_CN"  
  }, {  
    "active" : true,  
    "name" : "fr"  
  } ]  
}
```

Usage

Salesforce Knowledge must be enabled in your organization. This resource can be used in API version 31.0 and later. It retrieves the Knowledge language settings, including the default knowledge language and a list of supported Knowledge language information.

Retrieve a Version of an Article

Retrieves the version ID of an article.

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

Syntax

URI

```
/services/data/v25.0/knowledgeManagement/articleVersions/masterVersions/<articleVersionId>
```

Formats

JSON, XML

HTTP Method

GET

Authentication

Authorization: OAuth *access_token*

Parameters

None

Input

None required

Search for Metadata Elements of a Master Version

Searches for metadata elements of the online master version of an article.

Syntax

URI

```
/services/data/v25.0/knowledgeManagement/articleVersions/masterVersions?
filterArticleId=value1&FilterPublishStatus=value2
```

Formats

JSON, XML

HTTP Method

GET

Authentication

Authorization: OAuth *access_token*

Example

This example searches for the online version of the master article 'kA0x5000000jsh':

```
/services/data/v25.0/knowledgeManagement/articleVersions/masterVersions?
filterArticleId=kA0x5000000jsh&filterPublishStatus=online"
```

Search for Metadata Elements of a Translated Version

Searches for metadata elements of a translated version of an article.

Syntax

URI

```
/services/data/v25.0/knowledgeManagement/articleVersions/translations?filterArticleId=value1&filterLanguage=value2&FilterPublishStatus=value3";
```

Formats

JSON, XML

HTTP Method

GET

Authentication

Authorization: OAuth *access_token*

Example

This example searches for the German online translation of the article 'kA0x5000000jsh':

```
/services/data/v25.0/knowledgeManagement/articleVersions/translations?filterArticleId=kA0x5000000jsh&filterLanguage=de&filterPublishStatus=online"
```

Set a Translated Article Version to Complete

Sets a translated article version to complete.

Syntax

URI

```
/services/data/v25.0/knowledgeManagement/articleVersions/translations/<translationVersionID>
```

Formats

JSON, XML

HTTP Method

PATCH

Authentication

Authorization: OAuth *access_token*

Parameters

Parameter	Description
complete	Set this value to <code>true</code> for translations that are complete.

Input:

```
{
  "complete": "true"
}
```

Output:

HTTP status code 204 is returned when an existing record is updated.

Set a Translated Article Version to Incomplete

Sets a translated article version to incomplete.

Syntax

URI

```
/services/data/v25.0/knowledgeManagement/articleVersions/translations/<translationVersionID>
```

Formats

JSON, XML

HTTP Method

PATCH

Authentication

Authorization: OAuth *access_token*

Parameters

Parameter	Description
<code>complete</code>	Set this value to <code>false</code> to set a translation to incomplete.

Input:

```
{
  "complete": "false"
}
```

Output:

HTTP status code 204 is returned when an existing record is updated.

Submit an Article for Translation

Submits an article for translation for multiple languages and blocks translations for inactive languages.

Syntax

URI

```
/knowledgeManagement/articleVersions/translations
```

Available since release

25.0

Some parameters are supported in API versions 43.0 and higher.

Formats

JSON, XML

HTTP Method

POST

AuthenticationAuthorization: OAuth *access_token***Parameters**

Parameter	Description	Available
articleID	String. Required. The ID of the article.	25.0
assignments	JSON Array. An array of assignment details. Each assignment is a list with these properties: <ul style="list-style-type: none"> language: Required. A language code. assigneeId: A user Id or queue Id. dueDate: The date that the task is due. If you use this parameter, don't use the language parameter.	43.0
language	String. A language code . If you use this parameter, don't use the assignments parameter.	25.0-42.0
sendEmailNotification	Boolean. Specifies whether to send an email to the assignee (true) or not (false).	43.0

Example Request Body

```
{
  "articleId": "kA0xx00000000BO",
  "assignments": [
    {
      "language": "sq",
      "assigneeId": "005xx000001T7MF",
      "dueDate": ""
    },
    {
      "language": "zh_CN",
      "assigneeId": "005xx000001T7MF",
      "dueDate": ""
    }
  ],
  "sendEmailNotification": true
}
```

Unpublish the Master Version of an Article

Unpublishes the online master version of an article when there isn't an existing draft article.

The publishing actions are defined by the field change you request on the resource. To unpublish the master version, use "publishStatus": "draft".

Syntax

URI

```
/services/data/v25.0/knowledgeManagement/articleVersions/masterVersions/<versionId>
```

Formats

JSON, XML

HTTP Method

PATCH

Authentication

Authorization: OAuth *access_token*

Parameters

Parameter	Description
publishStatus	Publishing status of the article. Use <code>draft</code> .

Input

```
{
  "publishStatus": "draft"
}
```

Output:

HTTP status code 204 is returned when an existing record is updated.

Unpublish the Online Version of a Translated Article

Unpublishes the online version of a translated article.

The publishing actions are defined by the field change you request on the resource. To edit and remove a translation from online state, use `"publishStatus": "draft"`.

Syntax

URI

```
/services/data/v25.0/knowledgeManagement/articleVersions/translations/<translationVersionID>
```

Formats

JSON, XML

HTTP Method

PATCH

Authentication

Authorization: OAuth *access_token*

Parameters

Parameter	Description
publishStatus	Publishing status of the article. Use <code>draft</code> .

Input:

```
{
  "publishStatus": "draft"
}
```

Parameterized Search

Executes a simple RESTful search using parameters instead of a SOSL clause. Indicate parameters in a URL in the GET method. Or, use POST for more complex JSON searches.

Syntax

URI

`/services/data/v $XX.X$ /parameterizedSearch/?q=searchString`

Formats

JSON, XML


HTTP Method

GET, POST

Authentication

Authorization: Bearer *token*

Required Global Parameters

Name	Description
q	<p>A search string that is properly URL-encoded.</p> <p> Note: SOSL clauses aren't supported.</p> <p>Available in version 36.0 and later.</p>

Optional Global Parameters

Name	Type	Supported Methods	Description
dataCategory	string	GET	<p>Single value. If an organization uses Salesforce Knowledge articles or answers, <code>dataCategory</code> filters all search results based on one data category.</p> <p>For example, <code>dataCategory=GlobalCategory__c</code> below <code>NorthAmerica__c</code>.</p>

Name	Type	Supported Methods	Description
			<p>When using <code>dataCategories</code>, specify a Salesforce Knowledge article or answer type with <code>subject</code> and all the required parameters.</p> <p>For example:</p> <pre>q=tourism&subject=KnowledgeArticleVersion&KnowledgeArticleVersion.where=language='en_US'+and+publishStatus='online'&KnowledgeArticleVersion.fields=id,title&dataCategory=Location__c+Below+North_America__c</pre> <p>If you require multiple <code>dataCategory</code> filters, use <code>dataCategories</code> with the POST method.</p>
<code>dataCategories</code>	<code>dataCategoriesFilter[]</code>	POST	<p>If an organization uses Salesforce Knowledge articles or answers, filters all search results based on one or more data categories.</p> <p>When using <code>dataCategories</code>, specify a Salesforce Knowledge article or answer type with <code>subjects</code> and the required parameters.</p> <p>For example:</p> <pre>{ "q": "Acme", "fields": ["id", "title"], "subjects": [{"name": "KnowledgeArticleVersion", "where": "language='en_US' and publishstatus='draft'"}], "dataCategories": [{"groupName": "location__c", "operator": "below", "categories": ["North_America__c"]}] }</pre>
<code>defaultLimit</code>	string	GET, POST	<p>Single value. The maximum number of results to return for each <code>subject</code> (GET) or <code>subjects</code> (POST) specified.</p> <p>The maximum <code>defaultLimit</code> is 2000.</p> <p>At least one <code>subject</code> must be specified.</p> <p>GET example: <code>defaultLimit=10&subject=Account&subject=Contact</code>.</p> <p>When an <code>subject</code> limit is specified using <code>subject.limit=value</code>, such as <code>Account.limit=10</code>, this parameter is ignored for that object.</p>
<code>division</code>	string	GET, POST	<p>Single value. Filters search results based on the division field.</p> <p>For example in the GET method, <code>division=global</code>.</p> <p>Specify a division by its name rather than ID.</p> <p>All searches within a specific division also include the <code>global</code> division.</p>

Name	Type	Supported Methods	Description
fields	string	GET	<p>Comma-separated list of one or more fields to return in the response for each <code>subject</code> specified. At least one <code>subject</code> must be specified at the global level.</p> <p>For example: <code>fields=id&subject=Account&subject=Contact</code>.</p> <p>The global <code>fields</code> parameter is overridden when <code>subject</code> are specified using <code>subject.fields=field names</code>. For example, <code>Contact.fields=id,FirstName,LastName</code> would override the global setting of just returning the <code>id</code>.</p> <p>If unspecified, then the search results contain the IDs of records matching all fields for the specified object.</p> <p>Functions</p> <p>The following optional functions can be used within the <code>fields</code> parameter.</p> <ul style="list-style-type: none"> • <code>toLabel</code>: Translates response field value into the user's language. For example, <code>Lead.fields=id,toLabel(Status)</code>. This function requires extra setup. • <code>convertCurrency</code>: Converts response currency fields to the user's currency. For example, <code>Opportunity.fields=id,convertCurrency(Amount)</code>. This function requires extra setup. Multi-currency must be enabled for your org. • <code>format</code>: Applies localized formatting to standard and custom number, date, time, and currency fields. For example, <code>Opportunity.fields=id,format(Amount)</code>. <p>Aliasing is support within <code>fields</code> for <code>toLabel</code>, <code>convertCurrency</code>, and <code>format</code>. In addition, aliasing is required when the query includes the same field multiple times. For example, <code>Opportunity.fields=id,format(Amount) AliasAmount</code></p>
fields	string[]	POST	<p>Array of one or more fields to return in the response for each <code>subjects</code> specified. At least one <code>subjects</code> must be specified at the global level.</p> <p>For example:</p> <pre data-bbox="667 1415 1446 1713"> { "q": "Acme", "fields": ["Id", "Name", "Phone"], "subjects": [{"name": "Account"}, {"name": "Contact", "fields": ["Id", "FirstName", "LastName"]}, {"name": "Lead"}] } </pre> <p>The global <code>fields</code> parameter is overridden when <code>subjectsFilter[]</code> fields are specified. Such as, in the previous example, <code>Id</code>, <code>FirstName</code>, and <code>LastName</code> is returned for <code>Contact</code> instead of the global fields of <code>Id</code>, <code>Name</code> and <code>Phone</code>.</p>

Name	Type	Supported Methods	Description
			<p>If unspecified, then the search results contain the IDs of records matching all fields for the specified object.</p> <p>Functions</p> <p>The following optional functions can be used within the <code>fields</code> parameter.</p> <ul style="list-style-type: none"> <p><code>toLabel</code>: Translates response field value into the user's language. This function requires extra setup. For example:</p> <pre data-bbox="703 541 1446 747"> { ... "subjects": [{ "name": "Lead", "fields": ["Id", "toLabel (Status) "] }, ... } </pre> <p><code>convertCurrency</code>: Converts response currency fields to the user's currency. This function requires extra setup. Multi-currency must be enabled in the org. For example:</p> <pre data-bbox="703 884 1446 1089"> { ... "subjects": [{ "name": "Opportunity", "fields": ["Id", "convertCurrency (Amount) "] }] ... } </pre> <p><code>format</code>: Applies localized formatting to standard and custom number, date, time, and currency fields. For example:</p> <pre data-bbox="703 1188 1446 1394"> { ... "subjects": [{ "name": "Opportunity", "fields": ["Id", "format (Amount) "] }] ... } </pre> <p>Aliasing is supported within <code>fields</code> for <code>toLabel</code>, <code>convertCurrency</code>, and <code>format</code>. In addition, aliasing is required when the query includes the same field multiple times. For example:</p> <pre data-bbox="667 1535 1446 1740"> { ... "subjects": [{ "name": "Opportunity", "fields": ["Id", "format (Amount) AliasAmount"] }] ... } </pre>

Name	Type	Supported Methods	Description
<code>in</code>	string	GET, POST	<p>Scope of fields to search. If you specify one or more scope values, the fields are returned for all found objects.</p> <p>Use one of the following values:</p> <ul style="list-style-type: none"> • ALL • NAME • EMAIL • PHONE • SIDEBAR <p>This clause doesn't apply to articles, documents, feed comments, feed items, files, products, and solutions. If any of these objects are specified, the search is not limited to specific fields; all fields are searched.</p>
<code>metadata</code>	string	GET, POST	<p>Specifies if metadata should be returned in the response. No metadata is returned by default. To include metadata in the response, use the <code>LABELS</code> value, which returns the display label for the fields returned in search results. For example:</p> <pre>?q=Acme&metadata=LABELS</pre>
<code>networkIds</code>	string	GET	<p>Filters search results by a comma-separated list.</p> <p>A network ID represents the Experience Cloud site ID.</p>
<code>networkIds</code>	string[]	POST	<p>Filters search results by an array.</p> <p>A network ID represents the Experience Cloud site ID.</p>
<code>offset</code>	string	GET, POST	<p>Single value. The starting row offset into the result set returned.</p> <p>The maximum <code>offset</code> is 2000.</p> <p>Only one <code>subject</code> can be specified when using this parameter.</p>
<code>overallLimit</code>	string	GET, POST	<p>Single value. The maximum number of results to return across all <code>subject</code> parameters specified.</p> <p>The maximum <code>overallLimit</code> is 2000.</p>
<code>pricebookId</code>	string	GET, POST	<p>Single value. Filters product search results by a price book ID for only the Product2 object. The price book ID must be associated with the product that you're searching for. For example,</p> <pre>?q=laptop&subject=product2&pricebookId=01sxx0000002MffAAE</pre>
<code>snippet</code>	string	GET, POST	<p>The target length (maximum number of snippet characters) to return in Salesforce Knowledge article, case, case comment, feed, feed comment, idea, and idea comment search results. The <code>snippet</code> parameter displays contextual excerpts and highlights the search term for each article in the search results. Snippet results are used to differentiate matches to the search term in article search results. The target length can be from 50 to 1000 characters.</p>

Name	Type	Supported Methods	Description
			<p>Snippet and highlights are generated from email, text, and text area (long and rich) fields. Snippets aren't displayed for partially matching searches or if the user doesn't have access to the field that contains the snippet. Snippets are only displayed when 20 or fewer results are returned on a page.</p> <p>At least one of the following <code>subject</code> values must be specified.</p> <ul style="list-style-type: none"> To search a specific article type, use the article type name with the suffix <code>__kav</code>. To search all article types, use <code>KnowledgeArticleVersion</code>. To search case, case comment, feed, feed comment, idea, and idea comment types, use <code>Case</code>, <code>CaseComment</code>, <code>FeedItem</code>, <code>FeedComment</code>, <code>Idea</code>, and <code>IdeaComment</code>. <p>For example, <code>q=tourism&subject=Case&snippet=500</code>.</p>
<code>subject</code>	string	GET	<p>Objects to return in the response. Must be a valid object type.</p> <p>You can use multiple <code>subject</code> values, such as <code>subject=Account&subject=Contact</code>.</p> <p>If unspecified, then the search results contain the IDs of all objects.</p>
<code>subjects</code>	subjectsFilter[]	POST	<p>Objects to return in the response. Must contain valid object types. Use with the required parameters.</p> <p>For example:</p> <pre>{ "q": "Acme", "fields": ["id", "title"], "subjects": [{"name": "Solution__kav", "where": "language='en_US' and publishstatus='draft'"}, {"name": "FAQ__kav", "where": "language='en_US' and publishstatus='draft'"}] }</pre> <p>If unspecified, then the search results contain the IDs of all objects.</p>
<code>spellCorrection</code>	boolean	GET, POST	<p>Specifies whether spell correction is enabled for a user's search. When set to <code>true</code>, spell correction is enabled for searches that support spell correction. The default value is <code>true</code>.</p> <p>For example: <code>q=Acme&subject=Account&Account.fields=id&spellCorrection=true</code></p>
<code>updateTracking</code>	string	GET, POST	<p>Specifies a value of <code>true</code> to track keywords that are used in Salesforce Knowledge article searches only.</p> <p>If unspecified, the default value of <code>false</code> is applied.</p>

Name	Type	Supported Methods	Description
updateViewStat	string	GET, POST	Specifies a value of <code>true</code> to update an article's view statistics. Valid only for Salesforce Knowledge article searches. If unspecified, the default value of <code>false</code> is applied.

dataCategoriesFilter[] Parameters

Parameters must be specified in the order presented in the table (`groupName`, `operator`, `categories`).

Name	Type	Description
groupName	string	The name of the data category group to filter by.
operator	string	Valid values: <ul style="list-style-type: none"> • ABOVE • ABOVE_OR_BELOW • AT • BELOW
categories	string[]	The name of the categories to filter by.

subjectsFilter[] Parameters (POST Method Only)

Name	Type	Description
fields	string[]	Array of one or more fields to return in the response for the <code>subject</code> .
limit	string	Specify the maximum number of rows that are returned for the <code>subject</code> .
name	string	Name of the <code>subject</code> to return in the response.
orderBy	string	Controls the field order of the results using the following syntax <code>"orderBy" : "field {ASC DESC} [NULLS_{FIRST LAST}] "</code> For example: <pre>{ ... "subjects": [{"name": "Account", "fields": ["Id", "name"], "orderBy": "Name DESC Nulls_last"}] ... }</pre> <ul style="list-style-type: none"> • ASC: ascending. Default. • DESC: descending. • NULLS_FIRST: Null records at the beginning of the results. Default. • NULLS_LAST: Null records at the end of the results.

Name	Type	Description
where	string	Filter search results for this object by specific field values. For example, <code>where : conditionExpression</code> . Here the <code>conditionExpression</code> of the WHERE clause uses the following syntax: <code>fieldExpression [logicalOperator fieldExpression2 ...]</code> . Add multiple field expressions to a condition expression by using logical and comparison operators.

subject-level Parameters (GET Method Only)

The following optional parameters can be used with the `subject` parameter in a GET method to further refine search results. These settings would override any settings specified at the global level.

The format is `subject.parameter`, such as `Account.fields`. An `subject` must be specified to use these parameters, for example, `subject=Account&Account.fields=id, name`.

Name	Type	Description
fields	string	Comma-separated list of one or more fields to return in the response. For example, <code>KnowledgeArticleVersion.fields=id, title</code> .
limit	string	Specifies the maximum number of rows that are returned for the <code>subject</code> . For example, <code>Account.limit=10</code> .
orderBy	string	Controls the field order of the results using the following syntax <code>orderBy = field {ASC DESC} [NULLS_{FIRST LAST}]</code> For example: <code>Account.orderBy=Name</code> <ul style="list-style-type: none"> • ASC: ascending. Default. • DESC: descending. • NULLS_FIRST: Null records at the beginning of the results. Default. • NULLS_LAST: Null records at the end of the results.
where	string	Filter search results for this object by specific field values. Forexample, <code>Account.where = conditionExpression</code> . Here the <code>conditionExpression</code> of the WHERE clause uses the following syntax: <code>fieldExpression [logicalOperator fieldExpression2 ...]</code> . Add multiple field expressions to a condition expression by using logical and comparison operators. For example, <code>KnowledgeArticleVersion.where=publishstatus='online' and language='en_US'</code> .

Example GET Method

```
curl
https://MyDomainNamemy.salesforce.com/services/data/v55.0/parameterizedSearch/?q=Account&subject=Account&Account.fields=id,name&Account.limit=10
```

Example POST Method

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

Search

Executes the specified SOSL search. The search string must be URL-encoded.

For more information on SOSL see the [SOQL and SOSL Reference](#).

Syntax

URI

/services/data/v**XX.X**/search/?q=**SOSL_search_string**

Formats

JSON, XML

HTTP Method

GET

Authentication

Authorization: Bearer **token**

Parameters

Parameter	Description
q	A SOSL statement that is properly URL-encoded.

Example

See Search for a String.

Search Scope and Order

Returns an ordered list of objects in the default global search scope of a logged-in user. Global search keeps track of which objects the user interacts with and how often, and arranges the search results accordingly. Objects used most frequently appear at the top of the list

The returned list reflects the object order in the user's default search scope, including any pinned objects on the user's search results page. This call is useful if you want to implement a custom search results page using the optimized global search scope. The search string must be URL-encoded.

Syntax

URI

/services/data/v**XX.X**/search/scopeOrder

Formats

JSON, XML

HTTP Method

GET

Authentication

Authorization: Bearer **token**

Example

See Get the Default Search Scope and Order.

Search Result Layouts

Returns search result layout information for the objects in the query string. For each object, this call returns the list of fields displayed on the search results page as columns, the number of rows displayed on the first page, and the label used on the search results page.

This call supports bulk fetch for up to 100 objects in a query.

Syntax

URI

/services/data/v**XX.X**/search/layout/?q=**Comma_delimited_object_list**

Formats

JSON, XML

HTTP Method

GET

Authentication

Authorization: Bearer **token**

Response format

Property	Type	Description
field	String	Object and field name formatted with a period separating. For example: Account . Name.
format	String	The type of date field, such as the date only or date and time. Only date related types are specified; otherwise, null.
label	String	Name as it appears to users
name	String	API name

Example

Get Search Result Layouts for Objects

Search for Records Suggested by Autocomplete and Instant Results

Returns a list of suggested records whose names match the user's search string. The suggestions resource provides autocomplete results and instant results for users to navigate directly to likely relevant records, before performing a full search.

Syntax

URI

```
/services/data/vXX.X/search/suggestions?q=search_string&subject=object_types
```

Available since release

32.0

Formats

JSON, XML

HTTP methods

GET

Authentication

Authorization: Bearer **token**

Request body

None required

Request parameters


Parameter	Description
<code>fields</code>	Optional. Used for creating lookup queries. Specify multiple fields using a comma-separated list. Specifies which lookup fields to be returned in the response.
<code>dynamicFields</code>	Optional. Available in API version 48.0 and later. Used to return additional dynamic fields. Specify multiple options using a comma-separated list. For example, if <code>dynamicFields=secondaryField</code> then each suggested record in the results contains an additional field besides <code>Id</code> and <code>Name</code> (or <code>Title</code>) based on the next eligible field in the search layout.
<code>groupId</code>	Optional. Specifies one or more unique identifiers of one or more groups that the question to return was posted to. Specify multiple groups using a comma-separated list. This parameter is only applicable when the parameter type equals <code>question</code> . Don't use with the <code>userId</code> .
<code>ignoreUnsupportedSObjects</code>	Optional. If an unsupported object is included in a request, this parameter indicates what action to take. If it's set to <code>false</code> , an error is returned. If it's set to <code>true</code> , the object is ignored and no error is returned. See the Unsupported Objects section for reference. The default is <code>false</code> .
<code>limit</code>	Optional. Specifies the maximum number of suggested records to return. If a limit isn't specified, 5 records are returned by default. If there are more suggested records than the limit specified, the response body's <code>hasMoreResults</code> property is <code>true</code> .
<code>networkId</code>	Optional. Specifies one or more unique identifiers for the Experience Cloud sites to return the question to. Specify multiple sites using a comma-separated list. This


Parameter	Description
	parameter is only applicable when the parameter <code>type</code> equals <code>question</code> or parameter <code>subject</code> equals <code>user</code> .
<code>q</code>	Required. The user's search query string, properly URL-encoded. Suggestions are returned only if the user's query string meets the minimum length requirements: one character for queries in Chinese, Japanese, Korean, and Thai; three characters for all other languages. Query strings that exceed the maximum length of 255 characters (or 200 consecutive characters without a space break) return an error.
<code>subject</code>	<p>Required. The objects that the search is scoped to, such as <code>Account</code> or <code>offer__c</code>.</p> <p>If the <code>subject</code> value is <code>feedItem</code>, the <code>type</code> parameter is required and it must have a value of <code>question</code>.</p> <p>Specify up to 10 objects with a comma-separated list. For example: <code>subject=Account, Contact, Lead</code>. To take advantage of the feature, activate the CrossObjectTypeahead permission.</p> <p>To specify the specific fields to return by object, use the following syntax with multiple fields in a comma-separated list. The <code>subject</code> is lowercase.</p> <pre>subject=subject.fields=fields</pre> <p>For example:</p> <pre>&subject=Account, Contact, Lead&account.fields=Website, Phone &contact.fields=Phone</pre>
<code>topicId</code>	Optional. Specifies the unique identifier of the single topic that the question to return was tagged as. This parameter is only applicable when the parameter <code>type</code> equals <code>question</code> .
<code>type</code>	Required when the <code>subject</code> value is <code>feedItem</code> . Including this parameter for all other <code>subject</code> values doesn't affect the query. Specifies that the type of Feed is questions. Valid value: <code>question</code> .
<code>userId</code>	Optional. Specifies one or more unique identifiers of one or more users who authored the question to return. Specify multiple users using a comma-separated list. This parameter is only applicable when the parameter <code>type</code> equals <code>question</code> . Don't use with the <code>groupId</code> .
<code>useSearchScope</code>	<p>Optional. Available in API version 40.0 and later. The default value is <code>false</code>. If <code>false</code>, the objects specified in the request are used to suggest records. If <code>true</code>, in addition to the objects specified in the request, the user's search scope is used to suggest records. The search scope is the list of objects a user uses most frequently.</p> <ul style="list-style-type: none"> • If the request doesn't specify an object, use <code>useSearchScope=true</code>. • If <code>useSearchScope=true</code> and the user's search scope is empty, the default search scope is used to suggest records. • Typically, only the first 10 objects are used to suggest records. However, an admin can assign objects that are always considered when returning results. If configured, up to 15 objects are used to suggest records. For more information about assigning objects, see Assign Search Results Objects to Users (Beta).

Parameter	Description
	<ul style="list-style-type: none"> Objects specified in the <code>subject</code> parameter are prioritized over objects in the user's search scope. Values for the <code>ignoreUnsupportedSOjects</code> parameter aren't applied to the objects in the search scope. <p>This example uses only the search scope.</p> <pre data-bbox="617 451 1458 514">.../search/suggestions?q=Acme&useSearchScope=true</pre> <p>This example uses the search scope and the Account object.</p> <pre data-bbox="617 556 1458 619">.../search/suggestions?q=Acme&subject=Account&useSearchScope=true</pre>
where	<p>Optional. A filter that follows the same syntax as the SOQL <code>WHERE</code> clause. URLs encode the expression.</p> <p>Use the clause for an object, or globally for all compatible objects. An example of an object-specific clause is: <code>account.where=name%20LIKE%20%27Smith%25%27</code>. An example of a global clause is: <code>where=name%20LIKE%20%27Smith%25%27</code>. The parameter must be lower case. Any object-specific <code>where</code> clauses override the global <code>where</code> clause. You can't use this parameter for the Question object.</p> <p>To specify multiple entities, see the following example. This feature is available in version 38.0 and later.</p> <pre data-bbox="617 1029 1458 1669">...search/suggestions?q=Smith &subject=Account,Contact,KnowledgeArticleVersion,CollaborationGroup,Topic,FeedItem // Specifies a global where clause (to filter Account and Contact) &where=name%20LIKE%20%27Smith%25%27 // Overrides the global where clause for Knowledge Article (filtering by PublishStatus and Language is required for KnowledgeArticle) &knowledgearticleversion.where=PublishStatus='online'+and+language='en_US' // Overrides the global where clause for Topic &topic.where=networkid=<1234567891> // Overrides the global where clause for CollaborationGroup &collaborationgroup.where=networkid=<1234567891> // FeedItem-Question doesn't support where clauses, but we can filter the type and networkId &type=question &networkId==<1234567891></pre>

Usage

The suggestions resource returns records when the record's name field includes the exact text in the search string. The last term in the search string can match the beginning of a word. Records that contain the search string within a word aren't considered a match.

 **Note:** If the user's search query contains quotation marks or wildcards, those symbols are automatically removed from the query string in the URI.

 **Example:** The text string `national u` is treated as `national u*` and returns "National Utility", "National Urban Company", and "First National University".

Suggested Records Response

The suggestions resource returns display-ready data about likely relevant records that the user can access.

A relevance algorithm determines the order of results.

Each suggested record in the results contains these elements:

Element	Description
Attributes	The record's object type and the URL for accessing the record. Also includes the requested lookup fields' values. For example, if you requested <code>fields=Id,Name</code> , the result would include the ID and name.
Name (or Title)	The record's Name field. In the absence of a standard Name field, the Title field is used for these objects: <ul style="list-style-type: none"> • Dashboard • Idea • IdeaTheme • Note • Question In the absence of a standard Name or Title field, the main identifying field is used. For example, in cases, the Case Number is used.
Id	The record's unique identifier.

Example JSON Response Body

```
{
  "autoSuggestResults" : [ {
    "attributes" : {
      "type" : "Account",
      "url" : "/services/data/v55.0/subjects/Account/001xx000003DH6WAAW"
    },
    "Id" : "001xx000003DH6WAAW",
    "Name" : "National Utility Service"
  }, {
    {
      "attributes" : {
        "type" : "Account",
        "url" : "/services/data/v55.0/subjects/Account/001xx000003DHJ4AAO"
      },
      "Id" : "001xx000003DHJ4AAO",
      "Name" : "National Utility Service"
    }
  ]
}
```

```

    }, {
    {
      "attributes" : {
        "type" : "Account",
        "url" : "/services/data/v55.0/subjects/Account/001xx000003DHscAAG"
      },
      "Id" : "001xx000003DHscAAG",
      "Name" : "National Urban Technology Center"
    } ],
    "hasMoreResults" : false,
    "meta" : {
      "nameFields" : [ {
        "entityApiName" : "Account",
        "fieldApiName" : "Name"
      } ],
      "secondaryFields" : [ ]
    }
  }
}

```

Example JSON Response Body for a Multiple Object Request

```

{
  "autoSuggestResults" : [ {
    "attributes" : {
      "type" : "Account",
      "url" : "/services/data/v55.0/subjects/Account/001xx000003DMEKAA4"
    },
    "Id" : "001xx000003DMEKAA4",
    "Name" : "Joe Doe Printing"
  }, {
    {
      "attributes" : {
        "type" : "Account",
        "url" : "/services/data/v55.0/subjects/Account/001xx000003DLjvAAG"
      },
      "Id" : "001xx000003DLjvAAGO",
      "Name" : "Joe Doe Plumbing"
    }, {
      {
        "attributes" : {
          "type" : "Contact",
          "url" : "/services/data/v55.0/subjects/Contact/003xx000004U9Y9AAK"
        },
        "Id" : "003xx000004U9Y9AAK",
        "Name" : "John Doe"
      } ],
    "hasMoreResults" : false,
    "meta" : {
      "nameFields" : [ {
        "entityApiName" : "Account",
        "fieldApiName" : "Name"
      } ], {
        "entityApiName" : "Contact",

```



```

    "fieldApiName" : "Name"
  } ],
  "secondaryFields" : [ ]
}
}

```

Example XML Response Body

```

<?xml version="1.0" encoding="UTF-8"?
<suggestions>
  <autoSuggestResults type="Account"
url="/services/data/v55.0/subjects/Account/001xx000003DH6WAAW">
    <Id>001xx000003DH6WAAW</Id>
    <Name>National Utility Service</Name>
  </autoSuggestResults>
  <autoSuggestResults type="Account"
url="/services/data/v55.0/subjects/Account/001xx000003DHJ4AAO">
    <Id>001xx000003DHJ4AAO</Id>
    <Name>National Utility Service</Name>
  </autoSuggestResults>
  <autoSuggestResults type="Account"
url="/services/data/v55.0/subjects/Account/001xx000003DHscAAG">
    <Id>001xx000003DHscAAG</Id>
    <Name>National Urban Technology Center</Name>
  </autoSuggestResults>
  <hasMoreResults>true</hasMoreResults>
  <meta>
    <nameFields>
      <entityApiName>Account</entityApiName>
      <fieldApiName>Name</fieldApiName>
    </nameFields>
    <nameFields>
      <entityApiName>ContentDocument</entityApiName>
      <fieldApiName>Title</fieldApiName>
    </nameFields>
  </meta>
</suggestions>

```

Unsupported Objects

The suggestions resource supports all searchable objects except the following.

- ContentNote
- Event
- External objects
- FeedComment
- FeedPost
- IdeaComment
- Pricebook2
- Reply

- TagDefinition
- Task

Search Suggested Article Title Matches

Returns a list of Salesforce Knowledge article titles that match the user's search query string. Provides a shortcut to navigate directly to likely relevant articles before the user performs a search.

Syntax

URI

```
/services/data/vXX.X/search/suggestTitleMatches?q=searchString
&language=articleLanguage&publishStatus=articlePublicationStatus
```

Available since release

30.0

Formats

JSON, XML

HTTP methods

GET

Authentication

Authorization: Bearer *token*

Request body

None required

Request parameters

Parameter	Description
<code>articleTypes</code>	Optional. Three-character ID prefixes indicating the desired article types. You can specify multiple values for this parameter in a single REST call, by repeating the parameter name for each value. For example, <code>articleTypes=ka0&articleTypes=ka1</code> .
<code>categories</code>	Optional. The name of the data category group and name of the data category for desired articles, expressed as a JSON mapping. You can specify multiple data category group and data category pairs in this parameter. For example, <code>categories={"Regions":"Asia","Products":"Laptops"}</code> . Characters in the URL might need to be encoded. For this example, <code>categories=%7B%22Regions%22%3A%22Asia%22%2C%22Products%22%3A%22Laptops%22%7D</code> .
<code>channel</code>	Optional. The channel where the matching articles are visible. Valid values: <ul style="list-style-type: none"> • <code>AllChannels</code>—Visible in all channels the user has access to • <code>App</code>—Visible in the internal Salesforce Knowledge application • <code>Pkb</code>—Visible in the public knowledge base • <code>Csp</code>—Visible in the Customer Portal • <code>Prm</code>—Visible in the Partner Portal If <code>channel</code> isn't specified, the default value is determined by the type of user.

Parameter	Description
	<ul style="list-style-type: none"> • Pkb for a guest user • Csp for a Customer Portal user • Prm for a Partner Portal user • App for any other type of user <p>If <code>channel</code> is specified, the specified value may not be the actual value requested, because of certain requirements.</p> <ul style="list-style-type: none"> • For guest, Customer Portal, and Partner Portal users, the specified value must match the default value for each user type. If the values don't match or <code>AllChannels</code> is specified, then <code>App</code> replaces the specified value. • For all users other than guest, Customer Portal, and Partner Portal users: <ul style="list-style-type: none"> – If <code>Pkb</code>, <code>Csp</code>, <code>Prm</code>, or <code>App</code> are specified, then the specified value is used. – If <code>AllChannels</code> is specified, then <code>App</code> replaces the specified value.
<code>language</code>	Required. The language of the user's query. Specifies the language that matching articles are written in.
<code>limit</code>	Optional. Specifies the maximum number of articles to return. If there are more suggested articles than the limit specified, the response body's <code>hasMoreResults</code> property is <code>true</code> .
<code>publishStatus</code>	Required. The article's publication status. Valid values: <ul style="list-style-type: none"> • <code>Draft</code>—Articles aren't published in Salesforce Knowledge. • <code>Online</code>—Articles are published in Salesforce Knowledge. • <code>Archived</code>—Articles aren't published and are available in Archived Articles view.
<code>q</code>	Required. The user's search query string, properly URL-encoded. Suggestions are returned only if the user's query string meets the minimum length requirements: one character for queries in Chinese, Japanese, and Korean, and three characters for all other languages. Query strings exceeding the maximum length of 250 characters return an error.
<code>topics</code>	Optional. The topic of the returned articles. For example: <code>topics=outlook&topics=email</code> .
<code>validationStatus</code>	Optional. The validation status of returned articles.

Example for getting suggested articles with matching titles

```
curl https://MyDomainName.my.salesforce.com/services/data/v55.0/search/suggestTitleMatches?q=orange+banana&language=en_US&publishStatus=Online -H "Authorization: Bearer token"
```

Example JSON response body


```
{
  "autoSuggestResults" : [ {
    "attributes" : {
      "type" : "KnowledgeArticleVersion",
      "url" : "/services/data/v55.0/subjects/KnowledgeArticleVersion/ka0D00000004CcQ"
    },
    "Id" : "ka0D00000004CcQ",
    "UrlName" : "orange-banana",
    "Title" : "orange banana",
    "KnowledgeArticleId" : "kA0D00000004Cfz"
  } ],
  "hasMoreResults" : false
}
```

Usage

Salesforce Knowledge must be enabled in your organization. The user must have the “View Articles” permission enabled. The articles suggested include only the articles the user can access, based on the data categories and article types the user has permissions to view.


The Suggest Article Title Matches resource is designed to return display-ready data about likely relevant articles. Articles are suggested if their titles contain the entire query string, except stopwords, such as “a,” “for,” and “the.”

For example, a search for *Backpacking for desert* returns the article, “Backpacking in the desert.”

 **Note:** Articles with titles that include stopwords from the query string, such as “Backpacking for desert survival” in this example, appear before matching articles with titles that don’t include the stopwords.

Stopwords at the end of the query string are treated as search terms.

A wildcard is automatically appended to the last token in the query string.

 **Note:** If the user’s search query contains quotation marks or wildcards, those symbols are automatically removed from the query string in the URI along with any other special characters.

If the number of suggestions returned exceeds the limit specified in the request, the end of the response contains a field called `hasMoreResults`. Its value is `true` if the suggestions returned are only a subset of the suggestions available, and `false` otherwise.

Search Suggested Queries

Returns a list of suggested searches based on the user’s query string text matching searches that other users have performed in Salesforce Knowledge. Provides a way to improve search effectiveness, before the user performs a search.

Syntax

URI

```
/services/data/vXX.X/search/suggestSearchQueries?q=searchString
&language=languageOfQuery
```

Available since release

30.0

Formats

JSON, XML

HTTP methods

GET

AuthenticationAuthorization: Bearer *token***Request body**

None required

Request parameters

Parameter	Description
<code>channel</code>	<p>Optional. Specifies the Salesforce Knowledge channel where the article can be viewed. Valid values:</p> <ul style="list-style-type: none"> • <code>AllChannels</code>—Visible in all channels the user has access to • <code>App</code>—Visible in the internal Salesforce Knowledge application • <code>Pkb</code>—Visible in the public knowledge base • <code>Csp</code>—Visible in the Customer Portal • <code>Prm</code>—Visible in the Partner Portal <p>If <code>channel</code> isn't specified, the default value is determined by the type of user.</p> <ul style="list-style-type: none"> • <code>Pkb</code> for a guest user • <code>Csp</code> for a Customer Portal user • <code>Prm</code> for a Partner Portal user • <code>App</code> for any other type of user <p>If <code>channel</code> is specified, the specified value may not be the actual value requested, because of certain requirements.</p> <ul style="list-style-type: none"> • For guest, Customer Portal, and Partner Portal users, the specified value must match the default value for each user type. If the values don't match or <code>AllChannels</code> is specified, then <code>App</code> replaces the specified value. • For all users other than guest, Customer Portal, and Partner Portal users: <ul style="list-style-type: none"> – If <code>Pkb</code>, <code>Csp</code>, <code>Prm</code>, or <code>App</code> are specified, then the specified value is used. – If <code>AllChannels</code> is specified, then <code>App</code> replaces the specified value.
<code>language</code>	Required. The language of the user's query.
<code>limit</code>	Optional. Specifies the maximum number of suggested searches to return. If there are more suggested queries than the limit specified, the response body's <code>hasMoreResults</code> property is <code>true</code> .
<code>q</code>	Required. The user's search query string, properly URL-encoded. Suggestions are returned only if the user's query string meets the minimum length requirements: one character for queries in Chinese, Japanese, and Korean, and three characters for all other languages. Query strings exceeding the maximum length of 250 characters return an error.

Example for getting suggested queries

```
curl https://MyDomainName.my.salesforce.com/services/data/v55.0/search/suggestSearchQueries?q=app&language=en_US -H "Authorization: Bearer token"
```

Example JSON response body

```
{
  "autoSuggestResults" : [ {
    "0" : "apple",
    "1" : "apple banana",
  } ],
  "hasMoreResults" : false
}
```

Usage

Salesforce Knowledge must be enabled in your organization.

Queries are suggested if they exactly match the query string text. The text string must be a prefix within the query; it's not considered a match if it appears within a word. For example, the text string `app` would return suggested queries `apple banana` and `banana apples` but not `pineapple`.

If the number of suggestions returned exceeds the limit specified in the request, the end of the response contains a field called `hasMoreResults`. Its value is `true` if the suggestions returned are only a subset of the suggestions available, and `false` otherwise.

If the user's search query contains quotation marks or wildcards, those symbols are automatically removed from the query string in the URI.

Support Knowledge with REST API

Knowledge Support REST APIs allow both authorized and guest users to retrieve the user's visible data categories and their associated articles.

Authenticated users need the `UserProfile.apiEnabled` permission, Knowledge enabled in the organization, read rights on the article type, and any other knowledge specific permission or preference that controls visibility to articles.

Guest users need the `Guest Access to the Support API` preference enabled on the relevant Site, Knowledge enabled in the organization, and read rights on the article type and article channel that controls the visibility for guest users.

Syntax

The root endpoint for all Knowledge support APIs that can be open to guest users.

Available since release

38.0

Method

GET

Formats

JSON, XML

Authentication

OAuth accesstoken

Endpoint`/services/data/v $XX.X$ /support`**HTTP headers****Accept:** Optional. Can be either `application/json` or `application/xml`.**Input**

None

Output

The following resources are related to Knowledge.

```
{
  "dataCategoryGroups" : "/services/data/v $XX.X$ /support/dataCategoryGroups",
  "knowledgeArticles" : "/services/data/v $XX.X$ /support/knowledgeArticles"
  :
}
```

Where `v $XX.X$` is the API version you requested.**IN THIS SECTION:**[Data Category Groups](#)

Get data category groups that are visible to the current user.

[Data Category Detail](#)

Gets data category details and the child categories by a given category.

[Articles List](#)

Get a page of online articles for the given language and category through either search or query.

[Articles Details](#)

Gets all online article fields, accessible to the user.

Data Category Groups

Get data category groups that are visible to the current user.

Syntax

Available since release

38.0

Method

GET

Formats

JSON, XML

Authentication

OAuth accesstoken

Endpoint`/services/data/v $XX.X$ /support/dataCategoryGroups`

HTTP headers

Accept: Optional. Can be either `application/json` or `application/xml`.

Accept-language: Optional. Language to translate the categories. Any ISO-639 language abbreviation, and an ISO-3166 country code subtag in the HTTP Accept-Language header. Only one language accepted. If no language specified, the non-translated labels are returned.

Input:

string `sObjectName`: Required. `KnowledgeArticleVersion` only.

boolean `topCategoriesOnly`: Optional. Defaults to true

- True returns only the top level categories.
- False returns the entire tree.

 **Note:** All the input parameters are case-sensitive.


Output:

A list of the active data category groups that are visible to the current user in the site context. Returns id, name, label, and their top level categories or the entire data category group tree that are visible to the current user. The labels must be translated to the given language if they are available.

- **Data Category Group List**

This payload lists the active root Data Category Groups that can be used in other requests to return the data categories and articles related to it.

```
{
  "categoryGroups": [ Data Category Group, ...],
}
```

 **Note:** Returns only the active groups that are associated to the given entity (by `sObjectName`). Only `KnowledgeArticleVersion` is supported.

- **Data Category Group**

This represents an individual data category group, and its root category.


```
{
  "name": String, // the unique name of the category group
  "label": String, // returns the translated version if it is available
  "objectUsage" : String, // currently only "KnowledgeArticleVersion" is available.

  "topCategories": [ Data Category Summary, ...]
}
```

- **Data Category Summary**

This provides a summary of data category information. The Summary and Detail responses share common properties, with the goal of providing only as much information as is necessary from associated resources.

```
{
  "name": String, // the unique name of the category
  "label": String, // returns the translated version if it is available
  "url": URL, // the url points to the data category detail API
  "childCategories": [ Data Category Summary, ...] // null if topCategoriesOnly is true
}
```


 **Note:** The URL property is a pre-calculated path to the unique resource representing this data category, in this case it is a [Data Category Detail](#) API.

Example

Input

```
/services/data/v55.0/support/dataCategoryGroups?sObjectName=KnowledgeArticleVersion
```

Output

```
{
  "categoryGroups" : [ {
    "label" : "Doc",
    "name" : "Doc",
    "objectUsage" : "KnowledgeArticleVersion",
    "topCategories" : [ {
      "childCategories" : null,
      "label" : "All",
      "name" : "All",
      "url" :
"/services/data/v55.0/support/dataCategoryGroups/Doc/dataCategories/All?sObjectName=KnowledgeArticleVersion"
    } ]
  }, {
    "label" : "Manual",
    "name" : "Manual",
    "objectUsage" : "KnowledgeArticleVersion",
    "topCategories" : [ {
      "childCategories" : null,
      "label" : "All",
      "name" : "All",
      "url" :
"/services/data/v55.0/support/dataCategoryGroups/Manual/dataCategories/All?sObjectName=KnowledgeArticleVersion"
    } ]
  } ]
}
```

Usage

Salesforce Knowledge must be enabled in your organization. This resource can be used in API version 38.0 and later. Use the language code format used in [Which Languages Does Salesforce Support?](#)

Only the user's visible data categories are returned. A user might be able to see several sub trees in the category group, therefore, the top categories that are visible to the user in each group are returned.

Data Category Detail

Gets data category details and the child categories by a given category.

Syntax

Available since release

38.0

Method

GET

Formats

JSON, XML

Authentication

OAuth accesstoken

Endpoint

`/services/data/vXX.X/support/dataCategoryGroups/group/dataCategories/category`

HTTP headers

Accept: Optional. Can be either `application/json` or `application/xml`.

Accept-language: Optional. Language to translate the categories. Any ISO-639 language abbreviation, and an ISO-3166 country code subtag in the HTTP Accept-Language header. Only one language accepted. If no language specified, the non-translated labels are returned.

Input:

string sObjectName: Required. KnowledgeArticleVersion only.

Output:

Details of the category and a list of child categories (name, label, etc.).

- **Data Category Detail**

Used for situations where the hierarchical representation of data categories is important. The child property contains a list of child data categories.

```
{
  "name": String, // the unique name of the category
  "label": String, // returns the translated version if it is available
  "url": URL,
  "childCategories": [ Data Category Summary, ...],
}
```

 **Note:** If the category isn't visible to the current user the return is empty.

Example

Input

```
/services/data/v55.0/support/dataCategoryGroups/Doc/dataCategories/All?sObjectName=KnowledgeArticleVersion
```

Output

```
{
  "childCategories" : [ {
    "childCategories" : null,
    "label" : "Help",
    "name" : "Help",
    "url" :
  }
```

```

"/services/data/v55.0/support/dataCategoryGroups/Doc/dataCategories/Help?sObjectName=KnowledgeArticleVersion"
  }, {
    "childCategories" : null,
    "label" : "QA",
    "name" : "QA",
    "url" :
"/services/data/v55.0/support/dataCategoryGroups/Doc/dataCategories/QA?sObjectName=KnowledgeArticleVersion"
  } ],
  "label" : "All",
  "name" : "All",
  "url" :
"/services/data/v55.0/support/dataCategoryGroups/Doc/dataCategories/All?sObjectName=KnowledgeArticleVersion"
}

```

Usage

Salesforce Knowledge must be enabled in your organization. This resource can be used in API version 38.0 and later. Use the language code format used in [Which Languages Does Salesforce Support?](#).

Articles List

Get a page of online articles for the given language and category through either search or query.

Syntax

Available since release

38.0

Method

GET

Formats

JSON, XML

Authentication

OAuth access token

Endpoint

`/services/data/vxx.x/support/knowledgeArticles`

HTTP headers

Accept: Optional. Can be either `application/json` or `application/xml`.

Accept-language: Required. The article must be an active language in the user's organization

- If the language code isn't valid, an error message is returned: "The language code is not valid or not supported by Knowledge."
- If the language code is valid, but not supported by Knowledge, then an error message is returned: "Invalid language code. Check that the language is included in your Knowledge language settings."

Input:

string `q`: Optional, Performs an SOSL search. If the query string is null, empty, or not given, an SOQL query runs.

The characters `?` and `*` are used for wildcard searches. The characters `(`, `)`, and `"` are used for complex search terms. See https://developer.salesforce.com/docs/atlas.en-us.soql_sosl.meta/soql_sosl/sforce_api_calls_sosl_find.htm.

string `channel`: Optional, defaults to user's context. For information on channel values, see [Valid channel values](#).


- **App**: Visible in the internal Salesforce Knowledge application
- **Pkb**: Visible in the public knowledge base
- **Csp**: Visible in the Customer Portal
- **Prm**: Visible in the Partner Portal

string `categories` in map json format `{"group1": "category1", "group2": "category2", ...}`)

Optional, defaults to None. Category group must be unique in each group:category pair, otherwise you get `ARGUMENT_OBJECT_PARSE_ERROR`. There is a limit of three data category conditions, otherwise you get `INVALID_FILTER_VALUE`.

string `queryMethod` values are: `AT`, `BELOW`, `ABOVE`, `ABOVE_OR_BELOW`. Only valid when categories are specified, defaults to `ABOVE_OR_BELOW`.

string `sort`: Optional, a sortable field name `LastPublishedDate`, `CreatedDate`, `Title`, `ViewScore`. Defaults to `LastPublishedDate` for query and relevance for search.

 **Note**: When sorting on `ViewScore` it is only available for query, not search, and no pagination is supported. You only get one page of results.

string `order`: Optional, either `ASC` or `DESC`, defaults to `DESC`. Valid only when `sort` is valid.

integer `pageSize`: Optional, defaults to 20. Valid range 1 to 100.

integer `pageNumber` : Optional, defaults to 1.

Output:

A page of online articles in the given language and category visible to the current user.

- **Article Page**

A page of articles. The individual entries are article summaries so the size can be kept at a minimum.

```
{
  "articles": [ Article Summary, ... ], // list of articles
  "currentPageUrl": URL, // the article list API with current page number
  "nextPageUrl": URL, // the article list API with next page number,
  // which can be null if there are no more articles.
  "pageNumber": Int // the current page number, starting at 1.
}
```

 **Note**: The API supports paging. Each page of responses includes a URL to its page, as well as the URL to the next page of articles.

 **Note**: if the user input parameter has the default value, the API does not show this parameter in either `currentPageUrl` or `nextPageUrl`.

- **Article Summary**

A summary of an article used in a list of article responses. It shares similar properties to the Article Detail representation, as one is a superset of the other.

```
{
  "id": Id, // articleId
  "articleNumber": String,
```

```

    "articleType": String, // apiName of the article type, available in API v44.0
and later
    "title": String,
    "urlName": String, // available in API v44.0 and later
    "summary": String,
    "url": URL, // to the Article Detail API
    "viewCount": Int, // view count in the interested channel
    "viewScore": double (in xxx.xxxx precision), // view score in the interested
channel.
    "upVoteCount": int, // up vote count in the interested channel.
    "downVoteCount": int, // down vote count in the interested channel.
    "lastPublishedDate": Date // last publish date in ISO8601 format
    "categoryGroups": [ Data Category Group, ... ]}

```

The "url" property always points to the [Article Details](#) resource endpoint. For information on valid channel values, see the [channel parameter description](#)

- **Data Category Group**

An individual data category group, its root category, and a list of selected data categories in the group.

```

{
    "groupName": String, // the unique name of the category group
    "groupLabel": String, // returns the translated version
    "selectedCategories": [ Data Category Summary, ... ]
}

```

- **Data Category Summary**

Provides a summary of data category information. The Summary and Detail responses share common properties.

```

{
    "categoryName": String, // the unique name of the category
    "categoryLabel": String, // returns the translated version, per the API
language specified
    "url": String // returns the url for the DataCategory REST API.
}

```



Note: The outputs of Data Category Group and Data Category Summary in Article List API are different from the Data Category Groups API.

Example

Input

```

/services/data/v55.0/support/knowledgeArticles?sort=ViewScore&channel=Pkb&pageSize=3
HTTP Headers:
Content-Type: application/json; charset=UTF-8
Accept: application/json
Accept-Language: en-US

```

Output

```

{
  "articles" : [ {
    "articleNumber" : "000001002",

```

```

"categoryGroups" : [ ],
"downVoteCount" : 0,
"id" : "kA0xx000000000BCAQ",
"lastPublishedDate" : "2015-02-25T02:07:18Z",
"summary" : "With this online Chinese input tool, you can type Chinese characters
through your web browser without installing any Chinese input software in your system.
The Chinese online input tool uses the popular Pin Yin input method. It is a fast and
convenient tool to input Chinese on English OS environments.",
"title" : "Long text test",
"upVoteCount" : 0,
"url" : "/services/data/v55.0/support/knowledgeArticles/kA0xx000000000BCAQ",
"viewCount" : 4,
"viewScore" : 100.0
}, {
"articleNumber" : "000001004",
"categoryGroups" : [ ],
"downVoteCount" : 0,
"id" : "kA0xx000000000LCAQ",
"lastPublishedDate" : "2016-06-21T21:11:02Z",
"summary" : "The number of characters required for complete coverage of all these
languages' needs cannot fit in the 256-character code space of 8-bit character encodings,
requiring at least a 16-bit fixed width encoding or multi-byte variable-length encodings.
\r\n\r\nAlthough CJK encodings have common character sets, the encodings often used to
represent them have been developed separately by different East Asian governments and
software companies, and are mutually incompatible. Unicode has attempted, with some
controversy, to unify the character sets in a process known as Han unification.\r\n\r\nCJK
character encodings should consist minimally of Han characters p",
"title" : "Test Images",
"upVoteCount" : 0,
"url" : "/services/data/v55.0/support/knowledgeArticles/kA0xx000000000LCAQ",
"viewCount" : 0,
"viewScore" : 0.0
}, {
"articleNumber" : "000001012",
"categoryGroups" : [ ],
"downVoteCount" : 0,
"id" : "kA0xx0000000006GCAQ",
"lastPublishedDate" : "2016-06-21T21:10:48Z",
"summary" : null,
"title" : "Test Draft 2",
"upVoteCount" : 0,
"url" : "/services/data/v55.0/support/knowledgeArticles/kA0xx0000000006GCAQ",
"viewCount" : 0,
"viewScore" : 0.0
} ],
"currentPageUrl" :
"/services/data/v55.0/support/knowledgeArticles?channel=Pkb&pageSize=3&sort=ViewScore",
"nextPageUrl" : null,
"pageNumber" : 1
}

```

Usage

Salesforce Knowledge must be enabled in your organization. This resource can be used in API version 38.0 and later. The Custom File Field is not supported because it returns a link to a binary stream. Use the language code format used in [Which Languages Does Salesforce Support?](#)

Valid `channel` Values

- When using the options string `channel`, where the matching articles are visible, the following values are valid.
 - `App`—Visible in the internal Salesforce Knowledge application
 - `Pkb`—Visible in the public knowledge base
 - `Csp`—Visible in the Customer Portal
 - `Ptm`—Visible in the Partner Portal
- If `channel` isn't specified, the default value is determined by the type of user.
 - `Pkb` for a guest user
 - `Csp` for a Customer Portal user
 - `Ptm` for a Partner Portal user
 - `App` for any other type of user
- If `channel` is specified, the specified value may be used to retrieve articles.
 - For guest, Customer Portal, and Partner Portal users, if the specified channel is other than the channel accessible to the user, an error is returned.
 - For all users other than guest, Customer Portal, and Partner Portal users, the specified channel value is used.

Articles Details

Gets all online article fields, accessible to the user.

Syntax

Available since release

38.0

Method

GET

Formats

JSON, XML

Authentication

OAuth access token

Endpoint

`/services/data/v $XX.X$ /support/knowledgeArticles/articleId`

`/services/data/v $XX.X$ /support/knowledgeArticles/articleUrlName` Available in API v44.0 and later

HTTP headers

Accept: Optional. Can be either `application/json` or `application/xml`.

Accept-language: Required. The article must be an active language in the user's organization

- If the language code isn't valid, an error message is returned: "The language code is not valid or not supported by Knowledge."
- If the language code is valid, but not supported by Knowledge, then an error message is returned: "Invalid language code. Check that the language is included in your Knowledge language settings."

Input:

string `channel`: Optional, defaults to user's context. For information on channel values, see [Valid channel Values](#).

- **App**: Visible in the internal Salesforce Knowledge application
- **Pkb**: Visible in the public knowledge base
- **Csp**: Visible in the Customer Portal
- **Prm**: Visible in the Partner Portal

boolean `updateViewStat`: Optional, defaults to true. If true, API updates the view count in the given channel as well as the total view count.

boolean `isUrlName`: Optional, defaults to false. If true, indicates that the last portion of the endpoint is a URL name instead of an article ID. Available in API v44.0 and later

Output:

The detailed fields of the article, if the article is online and visible to the current user.

- **Article Detail**

Full detail of an article, with complete metadata and layout-driven fields used for display of an article. It includes all the same properties as an [Article Summary](#) representation.

```
{
  "id": Id, // articleId,
  "articleNumber": String,
  "articleType": String, // apiName of the article type, available in API
v44.0 and later
  "title": String,
  "urlName": String, // available in API v44.0 and later
  "summary": String,
  "url": URL,
  "versionNumber": Int,
  "createdDate": Date, // in ISO8601 format
  "createdBy": User Summary on page 131,
  "lastModifiedDate": Date, // in ISO8601 format
  "lastModifiedBy": User Summary on page 131,
  "lastPublishedDate": Date, // in ISO8601 format
  "layoutItems": [ Article Field, ... ], // standard and custom fields visible
to the user, sorted based on the layouts of the article type.
  "categories": [ Data Category Groups, ... ],
  "appUpVoteCount": Int,
  "cspUpVoteCount": Int,
  "prmUpVoteCount": Int,
  "pkbUpVoteCount": Int,
  "appDownVoteCount": Int,
  "cspDownVoteCount": Int,
  "prmDownVoteCount": Int,
  "pkbDownVoteCount": Int,
  "allViewCount": Int,
  "appViewCount": Int,
```



```

    "cspViewCount": Int,
    "prmViewCount": Int,
    "pkbViewCount": Int,
    "allViewScore": Double,
    "appViewScore": Double,
    "cspViewScore": Double,
    "prmViewScore": Double,
    "pkbViewScore": Double
  }

```

- **User Summary**

```

{
  "id": String
  "isActive": boolean // true/false
  "userName": String // login name
  "firstName": String
  "lastName": String
  "email": String
  "url": String // to the chatter user detail url:
  /services/data/vXX.X/chatter/users/{userId}, for guest user, it will return null.
}

```

- **Article Field**

An individual field of article information, which is listed in an [Article Detail](#) in the order required by the administrator's layout.

```

{
  "type": Enum, // see the Notes
  "name": String, // In API v43.0 and earlier, the developer name. In
  API v44.0 and later, the API name.
  "label": String, // label
  "value": String,
}

```

Example

Input

```

/services/data/v55.0/support/knowledgeArticles/kA0xx000000000LCAQ
HTTP Headers:
Content-Type: application/json; charset=UTF-8
Accept: application/json
Accept-Language: en-US

```

Output

```

{
  "allViewCount" : 17,
  "allViewScore" : 100.0,
  "appDownVoteCount" : 0,
  "appUpVoteCount" : 0,
  "appViewCount" : 17,
  "appViewScore" : 100.0,
  "articleNumber" : "000001004",

```

```

"categoryGroups" : [ ],
"createdBy" : {
  "email" : "user@company.com",
  "firstName" : "Test",
  "id" : "005xx000001SvoMAAS",
  "isActive" : true,
  "lastName" : "User",
  "url" : "/services/data/v55.0/chatter/users/005xx000001SvoMAAS",
  "userName" : "admin@salesforce.org"
},
"createdDate" : "2016-06-21T21:10:54Z",
"cspDownVoteCount" : 0,
"cspUpVoteCount" : 0,
"cspViewCount" : 0,
"cspViewScore" : 0.0,
"id" : "kA0xx000000000LCAQ",
"lastModifiedBy" : {
  "email" : "user@company.com",
  "firstName" : "Test",
  "id" : "005xx000001SvoMAAS",
  "isActive" : true,
  "lastName" : "User",
  "url" : "/services/data/v55.0/chatter/users/005xx000001SvoMAAS",
  "userName" : "admin@salesforce.org"
},
"lastModifiedDate" : "2016-06-21T21:11:02Z",
"lastPublishedDate" : "2016-06-21T21:11:02Z",
"layoutItems" : [ {
  "label" : "Out of Date",
  "name" : "IsOutOfDate",
  "type" : "CHECKBOX",
  "value" : "false"
}, {
  "label" : "sample",
  "name" : "sample",
  "type" : "PICK_LIST",
  "value" : null
}, {
  "label" : "Language",
  "name" : "Language",
  "type" : "PICK_LIST",
  "value" : "en_US"
}, {
  "label" : "MyNumber",
  "name" : "MyNumber",
  "type" : "NUMBER",
  "value" : null
}, {
  "label" : "My File",
  "name" : "My_File",
  "type" : "FILE",
  "value" : null
} ],
"pkbDownVoteCount" : 0,

```

```

    "pkbUpVoteCount" : 0,
    "pkbViewCount" : 0,
    "pkbViewScore" : 0.0,
    "prmDownVoteCount" : 0,
    "prmUpVoteCount" : 0,
    "prmViewCount" : 0,
    "prmViewScore" : 0.0,
    "summary" : "The number of characters required for complete coverage of all these
languages' needs cannot fit in the 256-character code space of 8-bit character encodings,
requiring at least a 16-bit fixed width encoding or multi-byte variable-length encodings.
\r\n\r\nAlthough CJK encodings have common character sets, the encodings often used to
represent them have been developed separately by different East Asian governments and
software companies, and are mutually incompatible. Unicode has attempted, with some
controversy, to unify the character sets in a process known as Han unification.\r\n\r\nCJK
character encodings should consist minimally of Han characters p",
    "title" : "Test Images",
    "url" : "/services/data/v55.0/support/knowledgeArticles/kA0xx000000000LCAQ",
    "versionNumber" : 7
}

```

Usage

Salesforce Knowledge must be enabled in your organization. This resource can be used in API version 38.0 and later. The Custom File Field is not supported because it returns a link to a binary stream. Use the language code format used in [Which Languages Does Salesforce Support?](#).

A lookup custom field is visible to guest users depending on the lookup entity type. For example, User is visible, but Case and Account are not visible. Following standard fields are not visible to a guest user, even if they are in the layout:

- archivedBy
- isLatestVersion
- translationCompletedDate
- translationImportedDate
- translationExportedDate
- versionNumber
- visibleInInternalApp
- visibleInPKB
- visibleToCustomer
- visibleToPartner

Valid `channel` Values

- When using the options string `channel`, where the matching articles are visible, the following values are valid.
 - `App`—Visible in the internal Salesforce Knowledge application
 - `Pkb`—Visible in the public knowledge base
 - `Csp`—Visible in the Customer Portal
 - `Prm`—Visible in the Partner Portal
- If `channel` isn't specified, the default value is determined by the type of user.

- `Pkb` for a guest user
- `Csp` for a Customer Portal user
- `Ppm` for a Partner Portal user
- `App` for any other type of user
- If `channel` is specified, the specified value may be used to retrieve articles.
 - For guest, Customer Portal, and Partner Portal users, if the specified channel is other than the channel accessible to the user, an error is returned.
 - For all users other than guest, Customer Portal, and Partner Portal users, the specified channel value is used.

CHAPTER 4 Salesforce Knowledge Metadata API

In this chapter ...

- [ArticleType](#)
- [KnowledgeSettings](#)
- [DataCategoryGroup](#)
- [SearchSettings](#)
- [SearchLayouts](#)
- [SynonymDictionary](#)
- [ExternalDataSource](#)

You can manage articles, page layouts, and data categories with the Metadata API.

The main purpose of Metadata API is to move metadata between Salesforce orgs during the development process. Use Metadata API to deploy, retrieve, create, update, or delete customization information, such as custom object definitions and page layouts. Metadata API doesn't work directly with business data.

For generic Metadata API information see the [Metadata API Developer Guide](#).

ArticleType

Represents the metadata associated with an article type.

All articles in Salesforce Knowledge are assigned to an *article type*. An article's type determines the type of content it contains, its appearance, and which users can access it. For example, a simple FAQ article type can have two custom fields, `Question` and `Answer`, where article managers enter data when creating or updating FAQ articles. A more complex article type can have dozens of fields organized into several sections. Using layouts and templates, administrators can structure the article type in the most effective way for its particular content. User access to article types is controlled by permissions. For each article type, an administrator can grant "Create," "Read," "Edit," or "Delete" permissions to users. For example, the article manager can allow internal users to read, create, and edit FAQ article types, but let partner users only read FAQs. See "Knowledge Article Types" in the Salesforce online help and [Knowledge](#) in the *SOAP API Developer Guide*.

Declarative Metadata File Suffix and Directory Location

An ArticleType is defined as a custom object and is stored in the `objects` folder. ArticleTypes have a suffix `__kav` (instead of `__c` for custom objects). ArticleType field names have a suffix of `__c` like other custom objects, and must be dot-qualified with the name of the article type to which they belong. This is shown in the following sample `package.xml` file:

```
<?xml version="1.0" encoding="UTF-8"?>
<Package xmlns="http://soap.sforce.com/2006/04/metadata">
  <fullName>articlefilemetadata</fullName>
  <apiAccessLevel>Unrestricted</apiAccessLevel>
  <types>
    <members>newarticle__kav.description__c</members>
    <name>CustomField</name>
  </types>
  <types>
    <members>newarticle__kav</members>
    <name>CustomObject</name>
  </types>
</Package>
```

Version

ArticleTypes are available in API version 19.0 and later.

Fields

Field Name	Field Type	Description
articleTypeChannelDisplay	ArticleTypeChannelDisplay	Represents the article-type templates used to display an article in the various channels. See "Article Type Templates" in the Salesforce online help.
deploymentStatus	DeploymentStatus (enumeration of type string)	A string which represents the deployment status of a custom object or field. Valid values are: <ul style="list-style-type: none"> InDevelopment Deployed

Field Name	Field Type	Description
description	string	A description of the article type. Maximum of 1000 characters.
fields	CustomField[]	Represents one or more fields in the article type.
gender	Gender	Indicates the gender of the noun that represents the object. This is used for languages where words need different treatment depending on their gender.
label	string	Label that represents the object throughout the Salesforce user interface.
pluralLabel	string	Plural version of the label value.
startsWith	StartsWith (enumeration of type string)	Indicates whether the noun starts with a vowel, consonant, or is a special character. This is used for languages where words need different treatment depending on the first character. Valid values are listed in StartsWith.

ArticleTypeChannelDisplay

Determines the article-type templates that are used to display an article in its channels. Unless otherwise noted, all fields are createable, filterable, and nillable.

Field Name	Field Type	Description
articleTypeTemplates	ArticleTypeTemplate on page 137[]	Indicates which article-type template applies in the specified channel.

ArticleTypeTemplate

Sets the article-type template for a specific channel. If not specified, the default article-type template applies.

Field Name	Field Type	Description
channel	string	Specifies the channel where the article-type template applies: <ul style="list-style-type: none"> • AllChannels: all the available channels. • App: the Articles tab in Salesforce Knowledge. • Pkb: the public knowledge base. • Csp: the Customer Portal. • Ptm: the partner portal.
page	string	Represents the name of the custom Visualforce page used as a custom article-type template. Use this field when you select Page in the template field.
template	string	Indicates the article-type template used for the specified channel: <ul style="list-style-type: none"> • Page: custom Visualforce page. When specifying this value, you must also set the page field with the Visualforce page name. • Tab: display the sections you defined in the layout as tabs.

Field Name	Field Type	Description
		<ul style="list-style-type: none"> Toc: display the sections you defined in the layout as table of content.

Declarative Metadata Sample Definitions

A sample article type definition follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<CustomObject xmlns="http://soap.sforce.com/2006/04/metadata">
  <articleTypeChannelDisplay>
    <articleTypeTemplates>
      <channel>App</channel>
      <template>Tab</template>
    </articleTypeTemplates>
    <articleTypeTemplates>
      <channel>Prm</channel>
      <template>Tab</template>
    </articleTypeTemplates>
    <articleTypeTemplates>
      <channel>Csp</channel>
      <template>Tab</template>
    </articleTypeTemplates>
    <articleTypeTemplates>
      <channel>Pkb</channel>
      <template>Toc</template>
    </articleTypeTemplates>
  </articleTypeChannelDisplay>
  <deploymentStatus>Deployed</deploymentStatus>
  <description>Article type with custom fields</description>
  <fields>
    <fullName>description__c</fullName>
    <label>Description</label>
    <length>48</length>
    <type>Text</type>
  </fields>
  <label>newarticle</label>
  <pluralLabel>newarticles</pluralLabel>
</CustomObject>
```

Wildcard Support in the Manifest File

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

IN THIS SECTION:

[ArticleType Layout](#)

Represents the metadata associated with an article type page layout. Article type layouts determine which fields users can view and edit when entering data for an article, they also determine which sections appear when users view articles.

[ChannelLayout](#)

Represents the metadata associated with a communication channel layout. Communication channel layouts let admins share article content inline into communication channels (for example, in email publishers, Experience Builder sites, or social media publishers). Admins can create a list of fields for an article type or record type that they want to share for each communication channel. You can customize the order of the fields.

[ArticleType CustomField](#)

Represents the metadata associated with an article type custom field. Use this metadata type to create, update, or delete article type custom field definitions.

ArticleType Layout

Represents the metadata associated with an article type page layout. Article type layouts determine which fields users can view and edit when entering data for an article, they also determine which sections appear when users view articles.

The format of the article, for example whether layout sections display as subtabs or as a single page with links, is defined by the [article-type template](#). Each article type has only one layout, but you can choose a different template for each of the article type's four channels. See [Knowledge](#) in the *SOAP API Developer Guide*.

File Suffix and Directory Location

ArticleType layouts are stored in the `layouts` directory of the corresponding package directory. The prefix must match with the article type API name. The extension is `.layout`.

Version

ArticleType layouts are available in API version 19.0 and later.

Fields

Field Name	Field Type	Description
<code>layoutSections</code>	LayoutSection []	The main sections of the layout containing the article fields. The order here determines the layout order.

LayoutSection

LayoutSection represents a section of an ArticleType layout.

Field Name	Field Type	Description
<code>customLabel</code>	boolean	Indicates if this section's label is custom or standard (built-in). Custom labels can be any text, but must be translated. Standard labels have a predefined set of valid values, for example 'System Information', which are automatically translated.
<code>label</code>	string	The label; either standard or custom, based on the <code>customLabel</code> flag.

Field Name	Field Type	Description
layoutColumns	LayoutColumn[]	The columns of the layout, depending on the style. Salesforce Knowledge only supports one column in article type layouts.
style	LayoutSectionStyle (enumeration of type string)	The style of the layout. Salesforce Knowledge only supports the value <code>OneColumn</code> which displays a one-column page.

LayoutColumn

LayoutColumn represents the items in a column within a layout section.

Field Name	Field Type	Description
layoutItems	LayoutItem[]	The individual items within a column (ordered from top to bottom).

LayoutItem

LayoutItem represents the valid values that define a layout item.

Field Name	Field Type	Description
field	string	The field name reference, for example <code>MyField__c</code> .

Declarative Metadata Sample Definition

The following is the definition of an ArticleType page layout:

```
<?xml version="1.0" encoding="UTF-8"?>
<Layout xmlns="http://soap.sforce.com/2006/04/metadata">
  <layoutSections>
    <customLabel>true</customLabel>
    <label>Description</label>
    <layoutColumns>
      <layoutItems>
        <field>description__c</field>
      </layoutItems>
      <layoutItems>
        <field>dateTime__c</field>
      </layoutItems>
    </layoutColumns>
    <style>OneColumn</style>
  </layoutSections>
  <layoutSections>
    <label>Data Sheet</label>
    <layoutColumns>
      <layoutItems>
        <field>file__c</field>
      </layoutItems>
    </layoutColumns>
  </layoutSections>
</Layout>
```

```
<style>OneColumn</style>
</layoutSections>
</Layout>
```

ChannelLayout

Represents the metadata associated with a communication channel layout. Communication channel layouts let admins share article content inline into communication channels (for example, in email publishers, Experience Builder sites, or social media publishers). Admins can create a list of fields for an article type or record type that they want to share for each communication channel. You can customize the order of the fields.

File Suffix and Directory Location

Channel layout components have the suffix `.channelLayout` and are stored in the `channelLayouts` folder of the corresponding package directory. The prefix must match with the article type API name. In Lightning Knowledge, the prefix must match the API name for the knowledge object.

Version

Channel layout components are available in API version 32.0 and later.

Fields

Field Name	Field Type	Description
<code>doesExcludeFieldLabels</code>	boolean	Indicates whether field labels are excluded from the field contents in the communication channels where this layout applies (<code>true</code>) or not (<code>false</code>). The default is <code>false</code> , meaning field labels are inserted. Available when Lightning Knowledge is enabled in API version 48.0 and later.
<code>doesExcludeFiles</code>	boolean	Indicates whether related files are left off emails (<code>true</code>) or attached to emails (<code>false</code>). The default is <code>false</code> , meaning related files are attached. Available when Lightning Knowledge is enabled in API version 48.0 and later.
<code>enabledChannels</code>	string[]	The communication channels where this layout applies. In API version 32.0 to 46.0, the only valid value is <code>Email</code> . When Lightning Knowledge is enabled in API version 47.0 and later, <code>Chat</code> , <code>Messaging</code> , and <code>Social</code> are added valid values.
<code>label</code>	string	Required. The label for this configuration.
<code>layoutItems</code>	ChannelLayoutItem on page 142[]	The article fields contained in the layout. The order here determines the field order.
<code>recordType</code>	string	The name of the record type that the channel layout applies to. The default is the primary record type. Available in API version 41.0 and later.

ChannelLayoutItem

Field Name	Field Type	Description
field	string	Required. Name of the field. The format is <i>ArticleTypeName.FieldName</i> or, in Lightning Knowledge, <i>KnowledgeBaseName.FieldName</i> .

Declarative Metadata Sample Definition

The following is an example of a ChannelLayout component.

```
<?xml version="1.0" encoding="UTF-8"?>
<ChannelLayout xmlns="http://soap.sforce.com/2006/04/metadata">
  <label>Layout for Email</label>
  <layoutItems>
    <field>Knowledge.Question</field>
  </layoutItems>
  <layoutItems>
    <field>Knowledge.Answer</field>
  </layoutItems>
  <enabledChannels>Email</enabledChannels>
  <enabledChannels>Social</enabledChannels>
  <enabledChannels>Chat</enabledChannels>
  <doesExcludeFiles>false</doesExcludeFiles>
  <doesExcludeFieldLabels>true</doesExcludeFieldLabels>
</ChannelLayout>
```

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

```
<?xml version="1.0" encoding="UTF-8"?>
<Package xmlns="http://soap.sforce.com/2006/04/metadata">
  <types>
    <members>*</members>
    <name>ChannelLayout</name>
  </types>
  <version>41.0</version>
</Package>
```

ArticleType CustomField

Represents the metadata associated with an article type custom field. Use this metadata type to create, update, or delete article type custom field definitions.

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

Always specify the full name whenever you create or update a custom field. For example, a custom field on a custom object:

```
MyArticleType__kav.MyCustomField__c
```

Declarative Metadata File Suffix and Directory Location

Custom fields are defined as part of the article type. ArticleType field names have a suffix of `__c` like other custom objects, and must be dot-qualified with the name of the article type to which they belong. See [ArticleType](#) for more information.

Retrieving Custom Fields on Custom or Standard Objects

When you retrieve a custom or standard object, you return everything associated with the object. However, you can also retrieve only the custom fields for an object by explicitly naming the object and fields in `package.xml`. The following definition in `package.xml` retrieves the files `objects/MyCustomObject__c.object`, `objects/Account.object__c.object`, and `objects/MyArticleType__kav.object`, each containing one custom field definition.


```
<types>
  <members>MyCustomObject__c.MyCustomField__c</members>
  <members>Account.MyCustomAccountField__c</members>
  <members>MyArticleType__kav.MyOtherCustomField__c</members>
  <name>CustomField</name>
</types>
```

Version

ArticleTypes custom fields are available in API version 19.0 and later.

Fields for ArticleType

Unless otherwise noted, all fields are createable, filterable, and nillable.

 **Note:** If you create a knowledge validation rule, the errors always display at the top of the page, even if you add it beside the field. Therefore, write the errors descriptively so authors know how to satisfy the validation rule. For example, identify which field is causing the error. The Salesforce Classic user interface does not support field level error messages for articles.

Field Name	Field Type	Description
<code>defaultValue</code>	string	If specified, represents the default value of the field. This field was deprecated in API version 48.0.
<code>deleteConstraint</code>	DeleteConstraint (enumeration of type string)	Provides deletion options for lookup relationships. Valid values are: <ul style="list-style-type: none"> <code>Cascade</code>—Deletes the lookup record as well as associated lookup fields. <code>Restrict</code>—Prevents the record from being deleted if it's in a lookup relationship. <code>SetNull</code>—This is the default. If the lookup record is deleted, the lookup field is cleared. For more information on lookup relationships, see "Object Relationships" in Salesforce Help.
<code>description</code>	string	Description of the field.

Field Name	Field Type	Description
formula	string	If specified, represents a formula on the field.
formulaTreatBlankAs	TreatBlanksAs (enumeration of type string)	Indicates how to treat blanks in a formula. Valid values are: BlankAsBlank and BlankAsZero.
fullName	string	Inherited from Metadata, this field is defined in the WSDL for this metadata type. It must be specified when creating, updating, or deleting. See <code>createMetadata()</code> to see an example of this field specified for a call. This value cannot be <code>null</code> .
inlineHelpText	string	Represents the content of field-level help. For more information, see "Define Field-Level Help" in Salesforce Help.
label	string	Label for the field. You cannot update the label for standard fields in Article Type such as Title, UrlName, Summary, etc.
length	int	Length of the field.
picklist	Picklist (Including Dependent Picklist)	(Deprecated. Use this field in API version 37.0 and earlier only. In later versions, use <code>valueSet</code> instead.) If specified, the field is a picklist, and this field enumerates the picklist values and labels.
referenceTo	string	If specified, indicates a reference this field has to another object.
relationshipLabel	string	Label for the relationship.
relationshipName	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 might be <code>YourObjects</code> .
required	boolean	Indicates whether the field requires a value on creation (<code>true</code>) or not (<code>false</code>).
type	FieldType	Required. Indicates the field type for the field. Valid values are: <ul style="list-style-type: none"> • <code>Checkbox</code> available in version 30.0 and later • <code>Currency</code> • <code>ArticleCurrency</code> • <code>Date</code> • <code>DateTime</code> • <code>Email</code> • <code>File</code> • <code>Formula</code> • <code>Html</code> • <code>Lookup</code> • <code>Number</code>

Field Name	Field Type	Description
		<ul style="list-style-type: none"> • Percent • Phone • Picklist • DependentPicklist • MultiselectPicklist • Text • TextArea • LongTextArea • URL
visibleLines	int	Indicates the number of lines displayed for the field.

Declarative Metadata Sample Definition

```
<?xml version="1.0" encoding="UTF-8"?>
<CustomObject xmlns="http://soap.sforce.com/2006/04/metadata">
  <fields>
    <fullName>Comments__c</fullName>
    <description>add your comments about this object here</description>
    <label>Comments</label>
    <length>32000</length>
    <type>LongTextArea</type>
    <visibleLines>30</visibleLines>
  </fields>
</CustomObject>
```

KnowledgeSettings

Represents the metadata used to manage settings for Salesforce Knowledge.

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

In the package manifest, all organization settings metadata types are accessed using the Settings name. See Settings for more details.

File Suffix and Directory Location

KnowledgeSettings values are stored in a single file named `Knowledge.settings` in the `settings` directory. The `.settings` files are different from other named components because there's only one settings file for each settings component.

Version

KnowledgeSettings is available in API version 27.0 and later.

Fields

Field Name	Field Type	Description
answers	KnowledgeAnswerSettings	Represents the metadata used to manage settings for Salesforce Knowledge and Answers.
cases	KnowledgeCaseSettings	Represents the metadata used to manage settings for Salesforce Knowledge and Cases.
defaultLanguage	string	Required. The default language for Salesforce Knowledge. Use the abbreviation for the language, for example, en_US for United States English.
enableChatterQuestionKBDeflection	boolean	Indicates whether tracking for case deflection via Chatter is enabled (<code>true</code>) or not (<code>false</code>).
enableCreateEditOnArticlesTab	boolean	Indicates whether users can create and edit articles on the articles tab (<code>true</code>) or not (<code>false</code>).
enableExternalMediaContent	boolean	Indicates whether connecting to external media is enabled (<code>true</code>) or not (<code>false</code>).
enableKbStandardSharing	boolean	Indicates whether standard Salesforce sharing is enabled (<code>true</code>) or not (<code>false</code>).
enableKnowledge	boolean	Indicates whether Salesforce Knowledge is enabled (<code>true</code>) or not (<code>false</code>). This field is <code>false</code> by default.
enableKnowledgeAgentContribution	boolean	Indicates whether a user can create an article from a case (<code>true</code>) or not (<code>false</code>). (Classic only)
enableKnowledgeArticleTextHighlights	boolean	Indicates whether text snippet highlights in Salesforce Knowledge search results are enabled (<code>true</code>) or not (<code>false</code>). This field is <code>true</code> by default. Available in API version 47.0 and later.
enableKnowledgeAnswersPromotion	boolean	Indicates whether a user can create an article from a reply (<code>true</code>) or not (<code>false</code>). (Classic Only)
enableKnowledgeCaseRL	boolean	Indicates whether creating a list of cases linked to an article is enabled (<code>true</code>) or not (<code>false</code>). (Classic Only)
enableKnowledgeKeywordAutoComplete	boolean	Indicates whether auto-complete for keywords is enabled (<code>true</code>) or not (<code>false</code>) when searching Salesforce Knowledge. This field is <code>true</code> by default. Available in API version 47.0 and later.

Field Name	Field Type	Description
enableKnowledgeTitleAutoComplete	boolean	Indicates whether auto-complete for article titles is enabled (<code>true</code>) or not (<code>false</code>) when searching Salesforce Knowledge. This field is <code>true</code> by default. Available in API version 47.0 and later.
enableLightningAutoLoadRichTextField	boolean	Indicates whether rich text fields are enabled for editing when an article loads in Lightning Knowledge (<code>true</code>) or not (<code>false</code>). This field is <code>false</code> by default. Available in API version 47.0 and later.
enableLightningKnowledge	boolean	Indicates whether Lightning Knowledge is enabled (<code>true</code>) or not (<code>false</code>).
languages	KnowledgeLanguageSettings	A list of languages enabled for Salesforce Knowledge.
showArticleSummariesCustomerPortal	boolean	Indicates whether article summaries appear in the Customer Portal (<code>true</code>) or not (<code>false</code>).
showArticleSummariesInternalApp	boolean	Indicates whether article summaries appear in the internal knowledge base (<code>true</code>) or not (<code>false</code>).
showArticleSummariesPartnerPortal	boolean	Indicates whether article summaries appear in the partner portal (<code>true</code>) or not (<code>false</code>).
showValidationStatusField	boolean	Indicates whether validation status appears on articles (<code>true</code>) or not (<code>false</code>).
suggestedArticles	KnowledgeSuggestedArticlesSettings	Represents the metadata used to manage settings for the case fields used to suggest articles for cases. Available in API version 37.0 and later.
votingEnabled	boolean	When <code>true</code> , enables users to vote for a product or feature that uses Vote, such as Articles in Knowledge. Available in API version 50.0 and later.

KnowledgeAnswerSettings

Represents the metadata used to manage settings for Salesforce Knowledge and Answers.

Field Name	Field Type	Description
assignTo	string	Specifies the username an article is assigned to from Answers.
defaultArticleType	string	The default article type for articles created from Answers. Uses the API name of the article type.

Field Name	Field Type	Description
enableArticleCreation	boolean	Indicates whether users can create articles from Answers (<code>true</code>) or not (<code>false</code>).

KnowledgeCaseSettings

Represents the metadata used to manage settings for Salesforce Knowledge and Cases.

Field Name	Field Type	Description
articlePDFCreationProfile	string	The profile used to create a PDF of an article from Cases.
articlePublicSharingSites	KnowledgeSitesSettings	Represents the metadata used to manage settings for Salesforce Knowledge and Sites.
articlePublicSharingCommunities	KnowledgeSitesSettings	Represents the metadata used to manage settings for Salesforce Knowledge and Experience Cloud sites.
articlePublicSharingSitesChatterAnswers	KnowledgeSitesSettings	Represents the metadata used to manage settings for Salesforce Knowledge and Sites with Chatter Answers.
assignTo	string	Specifies the username an article is assigned to from Cases.
customizationClass	string	Specifies the Apex class used for customization.
defaultContributionArticleType	string	The default article type for articles created from Cases.
editor	KnowledgeCaseEditor (enumeration of type string)	Indicates the rich text editor type. Valid values are: <ul style="list-style-type: none"> • <code>simple</code> • <code>standard</code>
enableArticleCreation	boolean	Indicates whether users can create articles from Cases (<code>true</code>) or not (<code>false</code>). Controls whether other fields on KnowledgeCaseSettings can be set.
enableArticlePublicSharingSites	boolean	Indicates whether articles can be shared via a public site (URL) from Cases (<code>true</code>) or not (<code>false</code>).
enableCaseDataCategoryMapping	boolean	Indicates whether Case Data Category mapping is enabled (<code>true</code>) or not (<code>false</code>).
useProfileForPDFCreation	boolean	Indicates whether a profile is used to create a PDF of an article from Cases (<code>true</code>) or not (<code>false</code>).

KnowledgeSitesSettings

Represents the metadata used to manage settings for Salesforce Knowledge and Sites.

Field Name	Field Type	Description
site	string[]	Specifies the site used for Salesforce Knowledge and Sites.

KnowledgeLanguageSettings

A list of languages enabled for Salesforce Knowledge. KnowledgeLanguageSettings is available in API version 28.0 and later.

Field Name	Field Type	Description
language	KnowledgeLanguage[]	Represents the metadata used to manage settings for the languages enabled for Salesforce Knowledge.

KnowledgeLanguage

Represents the metadata used to manage settings for the languages enabled for Salesforce Knowledge. KnowledgeLanguage is available in API version 28.0 and later.

Field Name	Field Type	Description
active	boolean	Indicates whether the language is enabled (<code>true</code>) or not (<code>false</code>).
defaultAssignee	string	The default assignee for articles in the language.
defaultAssigneeType	KnowledgeLanguageLookupValueType (enumeration of type string)	Indicates the default assignee type. Valid values are: <ul style="list-style-type: none"> User Queue
defaultReviewer	string	The default reviewer for articles in the language.
defaultReviewerType	KnowledgeLanguageLookupValueType (enumeration of type string)	Indicates the default reviewer type. Valid values are: <ul style="list-style-type: none"> User Queue
name	string	The code for the language name, for example: English is <code>en</code> . See "What languages does Salesforce support?" in the Salesforce Help for a list of supported languages and their codes.

KnowledgeSuggestedArticlesSettings

Represents the metadata used to manage settings for the articles suggested for cases, work orders, and work order line items. The Work Order and Work Order Line Item objects must be enabled in the org to use the associated fields.

Field Name	Field Type	Description
caseFields	KnowledgeCaseFieldsSettings	Represents a list of the case fields used to suggest articles for the case.
useSuggestedArticlesForCase	boolean	Indicates whether case content is used to suggest articles for cases (true) or not (false).
workOrderFields	KnowledgeWorkOrderFieldsSettings	Represents a list of the work order fields used to suggest articles for the work order.
workOrderLineItemFields	KnowledgeWorkOrderLineItemFieldsSettings	Represents a list of the work order line item fields used to suggest articles for the work order line item.

KnowledgeCaseFieldsSettings

Represents a list of the case fields used to suggest articles for the case. Available in API version 37.0 and later.

Field Name	Field Type	Description
field	KnowledgeCaseField[]	Specifies the names of the case fields used to suggest articles for the case.

KnowledgeCaseField

Represents the name of the case field used to suggest articles for the case. Available in API version 37.0 and later.

Field Name	Field Type	Description
name	string	Specifies the name of the case field used to suggest articles for the case.

KnowledgeWorkOrderFieldsSettings

Represents a list of the work order fields used to suggest articles for the work order. Available in API version 39.0 and later.

Field Name	Field Type	Description
field	KnowledgeWorkOrderField[]	Specifies the names of the work order fields used to suggest articles for the work order.

KnowledgeWorkOrderField

Represents the name of the work order field used to suggest articles for the work order. Available in API version 39.0 and later.

Field Name	Field Type	Description
name	string	Specifies the name of the work order field used to suggest articles for the work order.

KnowledgeWorkOrderLineItemFieldsSettings

Represents a list of the work order line item fields used to suggest articles for the work order line item. Available in API version 39.0 and later.

Field Name	Field Type	Description
field	KnowledgeWorkOrderLineItemField[]	Specifies the names of the work order line item fields used to suggest articles for the work order line item.

KnowledgeWorkOrderLineItemField

Represents the name of the work order line item field used to suggest articles for the work order line item. Available in API version 39.0 and later.

Field Name	Field Type	Description
name	string	Specifies the name of the work order line item field used to suggest articles for the work order line item.

Declarative Metadata Sample Definition

This is a sample Knowledge settings file.

```
<?xml version="1.0" encoding="UTF-8"?>
<KnowledgeSettings xmlns="http://soap.sforce.com/2006/04/metadata">
  <answers>
    <enableArticleCreation>>false</enableArticleCreation>
  </answers>
  <cases>
    <articlePDFCreationProfile>partner portal knowledge
profile</articlePDFCreationProfile>
    <articlePublicSharingSites>
      <site>KnowledgeSite</site>
      <site>PKB2Site</site>
      <site>ChatterAnswersSite</site>
    </articlePublicSharingSites>
    <articlePublicSharingSitesChatterAnswers>
      <site>ChatterAnswersSite</site>
    </articlePublicSharingSitesChatterAnswers>
    <assignTo>testall@kb.org</assignTo>
    <defaultContributionArticleType>Support</defaultContributionArticleType>
    <editor>simple</editor>
    <enableArticleCreation>true</enableArticleCreation>
    <enableArticlePublicSharingSites>true</enableArticlePublicSharingSites>
  </cases>
</KnowledgeSettings>
```

```

    <useProfileForPDFCreation>true</useProfileForPDFCreation>
  </cases>
  <defaultLanguage>ja</defaultLanguage>
  <enableCreateEditOnArticlesTab>true</enableCreateEditOnArticlesTab>
  <enableExternalMediaContent>true</enableExternalMediaContent>
  <enableKnowledge>true</enableKnowledge>
  <showArticleSummariesCustomerPortal>true</showArticleSummariesCustomerPortal>
  <showArticleSummariesInternalApp>true</showArticleSummariesInternalApp>
  <showArticleSummariesPartnerPortal>true</showArticleSummariesPartnerPortal>
  <showValidationStatusField>true</showValidationStatusField>
  <suggestedArticles>
    <caseFields>
      <field>
        <name>Subject</name>
      </field>
      <field>
        <name>SuppliedEmail</name>
      </field>
    </caseFields>
    <useSuggestedArticlesForCase>true</useSuggestedArticlesForCase>
  </suggestedArticles>
</KnowledgeSettings>

```


Wildcard Support in the Manifest File

The wildcard character * (asterisk) in the `package.xml` manifest file doesn't apply to metadata types for feature settings. The wildcard applies only when retrieving all settings, not for an individual setting. For details, see [Settings](#). For information about using the manifest file, see [Deploying and Retrieving Metadata with the Zip File](#).

DataCategoryGroup

Represents a data category group.

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

 **Warning:** Using Metadata API to deploy category changes from one organization to another permanently removes categories and record categorizations that are not specified in your XML file. Salesforce recommends that you manually create data categories and record associations in an organization from Setup by entering *Data Categories* in the **Quick Find** box, then selecting **Data Categories** rather than deploying changes from a sandbox to a production organization. For more information, see [Usage](#).

Data category groups are provided to:

- Classify and filter data.
- Share data among users.

Every data category group contains items or data categories that can be organized hierarchically.

The example below shows the `Geography` data category group and its data categories.

```


Geography
  Worldwide
    North America
      United States of America

```

```

Canada
Mexico
Europe
Asia

```

 **Note:** See "Work with Data Categories" in the Salesforce online help for more information on data category groups, data categories, parent and sub categories.

File Suffix and Directory Location

The file suffix is `.datacategorygroup`. There is one file for each data category group stored in the `datacategorygroups` folder in the corresponding package directory.

Version

Data category groups are available in API version 18.0 and later.



Fields

This metadata type contains the following fields:

Field Name	Field Type	Description
<code>active</code>	boolean	Required. The status of the category group. Indicates whether this category group is active, (<code>true</code>), or not active (<code>false</code>).
<code>dataCategory</code>	DataCategory on page 153	Required. The top-level category within the data category group.
<code>description</code>	string	The description of the data category group.
<code>fullName</code>	string	Required. The unique name of the data category group. When creating a data category group, the <code>fullName</code> field and the file name (without its suffix) must match. 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 the Metadata component.
<code>label</code>	string	Required. Label that represents the object in Salesforce.
<code>objectUsage</code>	ObjectUsage on page 154	The objects that are associated with the data category group.


DataCategory

Represents an item (or data category) in the data category group. A data category can recursively contain a list of other data categories.

Field Name	Field Type	Description
<code>dataCategory</code>	<code>DataCategory[]</code>	A recursive list of sub data categories. For example, a list of countries within a continent. You can create up to 100 categories in a data category group and have up to 5 levels in a data category group hierarchy.
<code>label</code>	<code>string</code>	Required. Label for the data category throughout the Salesforce user interface.
<code>name</code>	<code>string</code>	<p>Required. The developer name of the data category used as a unique identifier for API access. The name can only contain characters, letters, and the underscore (<code>_</code>) character, must start with a letter, and cannot end with an underscore or contain two consecutive underscore characters.</p> <p> Important: The value for this field is defined once and cannot be changed later.</p> <p> Warning: If you deploy a category group that already exists in an organization, any category that is not defined in the XML file is permanently removed from your organization. For more information see Usage.</p>

ObjectUsage

Represents the objects that can be associated with the data category group. This association allows the object to be classified and filtered using the data categories.

Field Name	Field Type	Description
<code>object</code>	<code>string[]</code>	<p>A list of the object names that can be associated with the data category group. Valid values are:</p> <ul style="list-style-type: none"> <code>KnowledgeArticleVersion</code>—to associate articles. See "Modify Default Category Group Assignments for Articles" in the Salesforce online help for more information on data category groups association to articles. <code>Question</code>—to associate questions. You can associate the <code>Question</code> object with at most one category group. <p> Warning: If you deploy a category group that already exists in an organization, any object association that is not defined in the XML file is permanently removed from your organization. Ensure that your XML file specifies all the records associated with your category group in the organization. For more information see Usage.</p>

Declarative Metadata Sample Definition

This sample is the definition of the Geography data category group and its data categories:

```
<?xml version="1.0" encoding="UTF-8"?>
<DataCategoryGroup xmlns="http://soap.sforce.com/2006/04/metadata">
  <label>Geography</label>
  <description>Geography structure of service center locations</description>
  <fullName>geo</fullName>


  <dataCategory> <name>WW</name> <label>Worldwide</label>
    <dataCategory> <name>AMER</name> <label>North America</label>
      <dataCategory>
        <name>USA</name>
        <label>United States of America</label>
      </dataCategory>
      <dataCategory>
        <name>CAN</name>
        <label>Canada</label>
      </dataCategory>
      <dataCategory>
        <name>MEX</name>
        <label>Mexico</label>
      </dataCategory>
    </dataCategory>
    <dataCategory> <name>EMEA</name> <label>Europe, Middle East, Africa</label>
      <dataCategory>
        <name>FR</name>
        <label>France</label>
      </dataCategory>
      <dataCategory>
        <name>SP</name>
        <label>Spain</label>
      </dataCategory>
      <dataCategory>
        <name>UK</name>
        <label>United-Kingdom</label>
      </dataCategory>
    </dataCategory>
    <dataCategory>
      <name>APAC</name>
      <label>Asia</label>
    </dataCategory>
  </dataCategory>


  <objectUsage>
    <object>KnowledgeArticleVersion </object>
  </objectUsage>
</DataCategoryGroup>
```

Usage

When you deploy a category group XML file, Metadata API checks whether the category group exists in the target organization. If the category group does not exist, it is created. If the category group already exists, then Metadata API:

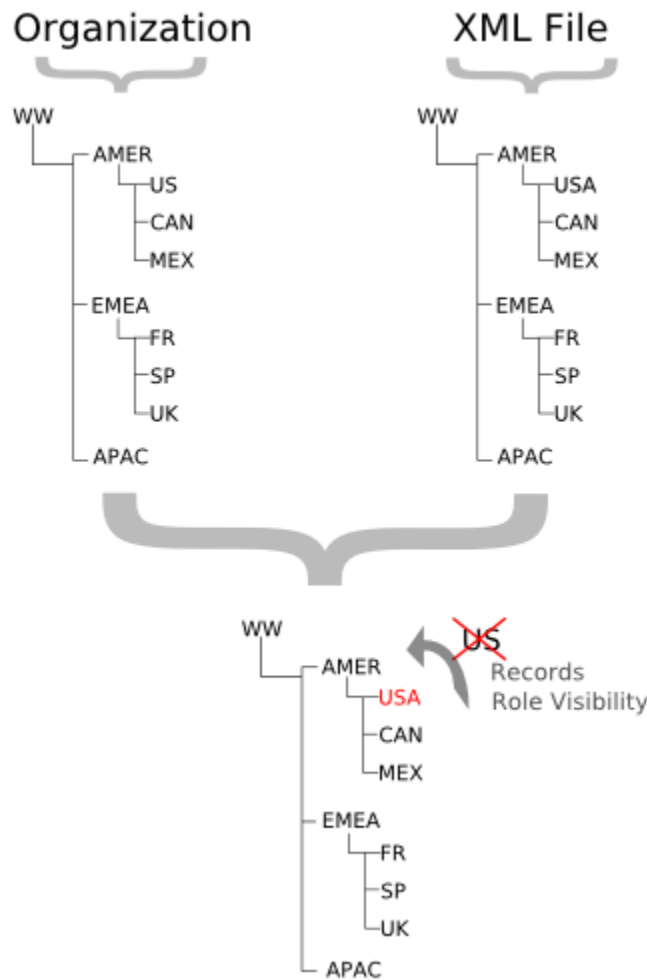
- Adds any new category or object defined in the XML file.
- Deletes any category that is not defined in the XML file. Records associated with the deleted categories are re-associated with the parent category.
- Deletes any object association that is not defined in the XML file.
- Moves any category if its hierarchical position differs from the position specified in the XML file.

 **Note:** When a category moves to a new parent category, users that have no visibility on the new parent category lose their visibility to the repositioned category.

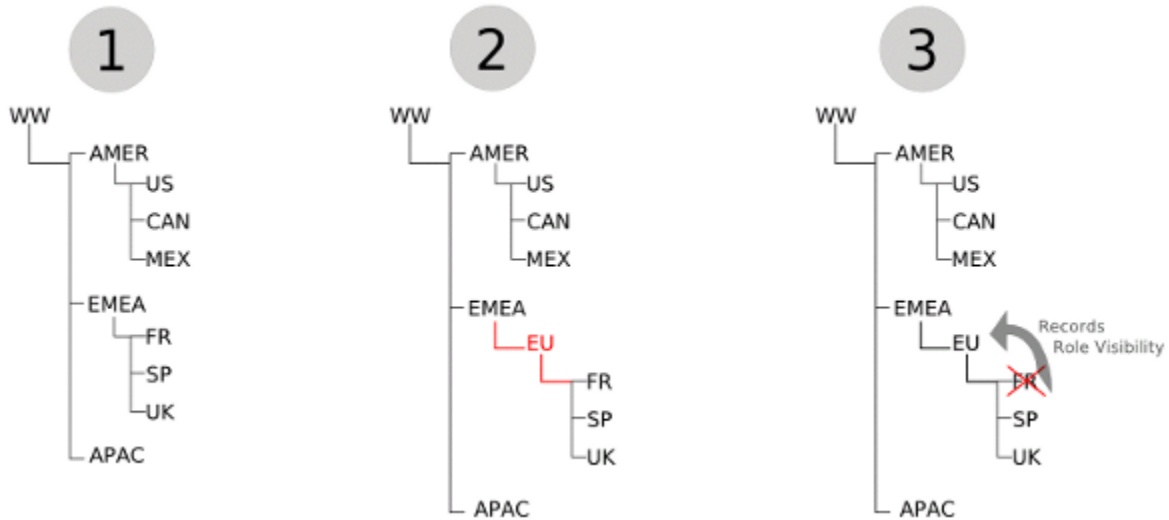
 **Note:** For more information about category deletion, category repositioning and its impact on record categorization and visibility see "Delete a Data Category" and "Modify and Arrange Data Categories" in the Salesforce online help.

Using Metadata API to deploy category changes from one organization to another permanently removes categories and record categorizations that are not specified in your XML file. Salesforce recommends that you manually create data categories and record associations in an organization from Setup by entering *Data Categories* in the *Quick Find* box, then selecting **Data Categories** rather than deploying changes from a sandbox to a production organization.

The following example illustrates what happens if you deploy an XML representation of a *Geography* data category group hierarchy to an organization that already has this data category group defined. Note that the organization contains a *US* category, while the XML file includes a *USA* category in the same hierarchical position. The Metadata API deployment process deletes the *US* category from the organization and moves associations for any records from *US* to the parent *AMER* category. It also adds the *USA* category under *AMER*. Note that all records that were previously categorized with *US* are now associated with the *AMER* category.



The next example illustrates what can happen when you delete or move a category in a data category group and deploy its XML representation from a sandbox to a production organization that already has this data category group defined. Hierarchy 1 shows the initial data category group in the sandbox organization. In hierarchy 2, we add an **EU** category under **EMEA** and move **FR**, **SP** and **UK** below **EU**. In hierarchy 3, we delete **FR** and associate its records with its new parent, **EU**. Finally, we deploy the changes from the sandbox to the production organization.



Metadata API has no concept of the order of the changes made to the sandbox organization. It just deploys the changes from one organization to another. During the deployment, it first notices the deletion of the `FR` category and removes it from the production organization. Consequently, it moves associations for any records from `FR` to its parent on the production organization, `EMEA`. Metadata API then adds the `EU` category and moves `SP` and `UK` below it. Although the category group hierarchy looks the same in both organizations, record categorization in production is different from the sandbox organization. The records that were originally associated with `FR` in hierarchy 1 are associated with `EU` in the sandbox organization, but are associated with `EMEA` in the production organization.

Wildcard Support in the Manifest File

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

SearchSettings

Represents an org's search settings. This type extends the Metadata metadata type and inherits its `fullName` field.

In the package manifest, all organization settings metadata types are accessed using the Settings name. See [Settings](#) for more details.

File Suffix and Directory Location

SearchSettings values are stored in a single file named `search.settings` in the `settings` folder. The `.settings` files are different from other named components because there's only one settings file for each settings component.

Version

SearchSettings is available in API version 37.0 and later.

Fields

Field Name	Field Type	Description
documentContentSearchEnabled	boolean	Indicates whether a full-text document search is performed.
enableAdvancedSearchInAlohaSidebar	boolean	Indicates whether advanced search is available in the search sidebar (<code>true</code>) or not (<code>false</code>). Available in Salesforce Classic only. Available in API version 46.0 and later.
enableEinsteinSearchPersonalization	boolean	Indicates whether search personalization is enabled (<code>true</code>) or not (<code>false</code>). Available in Lightning Experience only. Available in API version 47.0 and later.
enablePersonalTagging	boolean	Indicates whether users are allowed to group records from various objects by a common theme (<code>true</code>) or not (<code>false</code>). Personal tags are visible to the user only. Available in Salesforce Classic only. Available in API version 48.0 and later.
enablePublicTagging	boolean	Indicates whether users are allowed to group records from various objects by a common theme (<code>true</code>) or not (<code>false</code>). Personal tags are visible to all users. Available in Salesforce Classic only. Available in API version 48.0 and later.
enableSalesforceGeneratedSynonyms	boolean	Indicates whether search synonyms are enabled (<code>true</code>) or not (<code>false</code>). Available in API version 47.0 and later.
enableSearchTermHistory	boolean	Indicates whether users are allowed to group records from various objects by a common theme (<code>true</code>) or not (<code>false</code>). Public tags are visible to everyone in the organization. Available in Salesforce Classic only. Available in API version 48.0 and later.
enableSetupSearch	boolean	Indicates whether the search box in the Setup sidebar returns matching custom fields, custom objects, and other supported setup items when you press Enter (<code>true</code>) or not (<code>false</code>). The default is <code>true</code> in Developer, Performance, Professional, Enterprise, and Unlimited editions, and <code>false</code> in all other editions. Available in API version 47.0 and later.
enableSuggestArticlesLinksOnly	boolean	Indicates whether links are provided to knowledge articles from Cases similar to the current Case (<code>true</code>) or not (<code>false</code>). Available in API version 48.0 and later.
enableUseDefaultSearchEntity	boolean	Indicates whether to use the admin-specified default entity in sidebar search (<code>true</code>) or not (<code>false</code>). Available in Salesforce Classic only. Available in API version 48.0 and later.
optimizeSearchForCJKEnabled	boolean	Required. Indicates whether the search is optimized for the Japanese, Chinese, and Korean languages (<code>true</code>) or not (<code>false</code>). This setting affects sidebar search and the account search for Find Duplicates on a lead record in sidebar search and global search. Enable this option if users are searching

Field Name	Field Type	Description
		mostly in Japanese, Chinese, or Korean, and if the text in searchable fields is mostly in those languages.
recentlyViewedUsersForBlankLookupEnabled	boolean	Required. Indicates whether the list of records that are returned from a user autocomplete lookup and from a blank user lookup is taken from the user's recently viewed user records (<code>true</code>). Otherwise this setting is <code>false</code> if the lookup shows a list of recently accessed user records from across your org (<code>false</code>). Only applies to User object blank lookup searches.
searchSettingsByObject	SearchSettingsByObject	Required. Represents a list of search settings for each object.
sidebarAutoCompleteEnabled	boolean	Required. Indicates whether autocomplete is enabled for sidebar search (<code>true</code>) or not (<code>false</code>). Autocomplete is when users start typing search terms and sidebar search displays a matching list of recently viewed records.
sidebarDropDownListEnabled	boolean	Required. Indicates whether a dropdown list appears in the sidebar search section (<code>true</code>) or not (<code>false</code>). From this list, users can select to search within tags, within a specific object, or across all objects.
sidebarLimitToItemsIOwnCheckboxEnabled	boolean	Required. Indicates whether the Limit to Items I Own checkbox appears (<code>true</code>) or not (<code>false</code>). The checkbox allows your users to include only records for which they are the record owner when entering search queries in the sidebar.
singleSearchResultShortcutEnabled	boolean	Required. Indicates whether a shortcut is enabled (<code>true</code>) or not (<code>false</code>). With the shortcut, users skip the search results page and go directly to the record's detail page when their search returns only a single item. This setting doesn't apply to tags, case comments (in advanced search), and global search.
spellCorrectKnowledgeSearchEnabled	boolean	Required. Indicates whether spell check is enabled for Knowledge search (<code>true</code>) or not (<code>false</code>).

SearchSettingsByObject

Field Name	Field Type	Description
searchSettingsByObject	ObjectSearchSetting	Contains a list of search settings for each object.

ObjectSearchSetting

A list of search settings for each object.

Field Name	Field Type	Description
enhancedLookupEnabled	boolean	Required. Indicates whether enhanced lookups is enabled for the object (<code>true</code>) or not (<code>false</code>).
lookupAutoCompleteEnabled	boolean	Required. Indicates whether autocomplete is enabled for lookup search (<code>true</code>) or not (<code>false</code>). Autocomplete is when users edit the lookup field inline by choosing an autosuggestion.
name	string	Required. The entity name of the object being configured.
resultsPerPageCount	int	Required. The number of search results per page.

Declarative Metadata Sample Definition

The following is an example of the `Search.settings` file.

```
<?xml version="1.0" encoding="UTF-8"?>
  <SearchSettings xmlns="http://soap.sforce.com/2006/04/metadata">
    <enableSetupSearch>false</enableSetupSearch>
    <enableAdvancedSearchInAlohaSidebar>false</enableAdvancedSearchInAlohaSidebar>
    <enableQuerySuggestionPigOn>false</enableQuerySuggestionPigOn>
    <enableSalesforceGeneratedSynonyms>false</enableSalesforceGeneratedSynonyms>
    <enableSearchTermHistory>false</enableSearchTermHistory>
    <enablePublicTagging>false</enablePublicTagging>
    <enablePersonalTagging>false</enablePersonalTagging>
    <enableSuggestArticlesLinksOnly>false</enableSuggestArticlesLinksOnly>
    <enableUseDefaultSearchEntity>false</enableUseDefaultSearchEntity>
      <documentContentSearchEnabled>true</documentContentSearchEnabled>
      <optimizeSearchForCJKEEnabled>true</optimizeSearchForCJKEEnabled>
  </SearchSettings>
  <recentlyViewedUsersForBlankLookupEnabled>true</recentlyViewedUsersForBlankLookupEnabled>

  <searchSettingsByObject>
    <searchSettingsByObject>
      <enhancedLookupEnabled>false</enhancedLookupEnabled>
      <lookupAutoCompleteEnabled>false</lookupAutoCompleteEnabled>
      <name>Account</name>
      <resultsPerPageCount>25</resultsPerPageCount>
    </searchSettingsByObject>
    <searchSettingsByObject>
      <enhancedLookupEnabled>false</enhancedLookupEnabled>
      <lookupAutoCompleteEnabled>false</lookupAutoCompleteEnabled>
      <name>Activity</name>
      <resultsPerPageCount>25</resultsPerPageCount>
    </searchSettingsByObject>
    <searchSettingsByObject>
      <enhancedLookupEnabled>false</enhancedLookupEnabled>
      <lookupAutoCompleteEnabled>false</lookupAutoCompleteEnabled>
      <name>Asset</name>
      <resultsPerPageCount>25</resultsPerPageCount>
    </searchSettingsByObject>
  </searchSettingsByObject>
```

```

        <sidebarAutoCompleteEnabled>true</sidebarAutoCompleteEnabled>
        <sidebarDropDownListEnabled>true</sidebarDropDownListEnabled>

<sidebarLimitToItemsIOwnCheckboxEnabled>true</sidebarLimitToItemsIOwnCheckboxEnabled>
        <singleSearchResultShortcutEnabled>true</singleSearchResultShortcutEnabled>
        <spellCorrectKnowledgeSearchEnabled>true</spellCorrectKnowledgeSearchEnabled>

        <enableEinsteinSearchPersonalization>true</enableEinsteinSearchPersonalization>
    </SearchSettings>

```

Example Package Manifest

The following is an example package manifest used to deploy or retrieve the Search settings metadata for an organization.

```

<?xml version="1.0" encoding="UTF-8"?>
<Package xmlns="http://soap.sforce.com/2006/04/metadata">
    <types>
        <members>Search</members>
        <name>Settings</name>
    </types>
    <version>37.0</version>
</Package>

```

Wildcard Support in the Manifest File

The wildcard character * (asterisk) in the `package.xml` manifest file doesn't apply to metadata types for feature settings. The wildcard applies only when retrieving all settings, not for an individual setting. For details, see [Settings](#). For information about using the manifest file, see [Deploying and Retrieving Metadata with the Zip File](#).

SearchLayouts

Represents the metadata associated with the search layouts for an object. You can customize which fields to display for users in search results, search filter fields, lookup dialogs, and recent record lists on tab home pages. You can access SearchLayouts only by accessing its encompassing CustomObject.

For more information, see [Customize Layouts for Search Results](#) and [Customize Search Layouts for Custom Objects](#) in Salesforce Help.

Version

Search layouts for custom objects are available in API version 14.0 and later. The ability to modify search layouts for standard objects (except events and tasks) is available in API version 27.0 and later.

Fields

When defining metadata for search layouts:

- Any Name field defined as a text type is mandatory; it's always displayed as the first column in the search results page. When you query for a list of fields; the name field isn't returned but all other fields are. If you define the Name field as an autonumber type, it's not mandatory and you can remove it from the list. These rules apply to `customTabListAdditionalFields`,

lookupDialogsAdditionalFields, lookupPhoneDialogsAdditionalFields, and searchResultsAdditionalFields

- For custom objects, the search layout uses the API name, for example, MyCustomField__c instead of the field name My Custom Field.

Field	Field Type	Description
customTabListAdditionalFields	string[]	The list of fields displayed in the Recent <i>Object Name</i> list view for an object.
excludedStandardButtons	string[]	The list of standard buttons excluded from the search layout.
listViewButtons	string[]	The list of buttons available in list views for an object. This field is equivalent to the Buttons Displayed value in the <i>Object Name List View</i> in the related list of the object detail page in the UI. For more information, see “Standard and Enhanced Lookups in Salesforce Classic” in Salesforce Help.
lookupDialogsAdditionalFields	string[]	The list of fields displayed in a lookup dialog for the object. Salesforce objects often include one or more <i>lookup fields</i> that allow users to associate two records together in a relationship. For example, a contact record includes an <i>Account</i> lookup field that represents the relationship between the contact and the organization with which the contact is associated. A lookup search dialog helps you search for the record associated with the one being edited. Lookup filter fields allow you to filter your lookup search by a customized list of fields in the object. This field is equivalent to the <i>Lookup Dialogs</i> related list on the object detail page in the UI. For more information, see “Standard and Enhanced Lookups in Salesforce Classic” in Salesforce Help.
lookupFilterFields	string[]	The list of fields that can be used to filter enhanced lookups for an object. Enhanced lookups are optionally enabled by your administrator. This field is equivalent to the <i>Lookup Filter Fields</i> related list on the object detail page in the application user interface. For more information, see “Standard and Enhanced Lookups in Salesforce Classic” in Salesforce Help.
lookupPhoneDialogsAdditionalFields	string[]	The list of phone-related fields displayed in a lookup dialog for the object. This list enables integration of the fields with a softphone dial pad.

Field	Field Type	Description
		This field is equivalent to the <code>Lookup Phone Dialogs</code> related list on the object detail page in the application user interface.
<code>massQuickActions</code>	<code>string[]</code>	<p>The list of actions that you can use to perform mass quick action on records. Use this field to add an existing create or update action.</p> <p>You can perform mass quick actions on custom objects and all standard objects that support quick actions and have a search layout in Lightning Experience. This includes but isn't limited to cases, leads, accounts, campaigns, contacts, opportunities, and work orders.</p>
<code>searchFilterFields</code>	<code>string[]</code>	<p>The list of fields that can be used to filter a search for the object.</p> <p>This field is equivalent to the <code>Search Filter Fields</code> related list on the object detail page in the application user interface.</p>
<code>searchResultsAdditionalFields</code>	<code>string[]</code>	<p>The list of fields displayed in a search result for the object.</p> <p>This field is equivalent to the <code>Search Results</code> related list on the object detail page in the application user interface.</p>
<code>searchResultsCustomButtons</code>	<code>string[]</code>	The list of custom buttons available in a search result for the object. The actions associated with the buttons can be applied to any of the records returned in the search result.

Declarative Metadata Sample Definition

A sample definition of object's search layout is shown..

```
<?xml version="1.0" encoding="UTF-8"?>
  <CustomObject xmlns="http://soap.sforce.com/2006/04/metadata">
    . . .
    <searchLayouts>
      <listViewButtons>New</listViewButtons>
      <listViewButtons>Accept</listViewButtons>
      <listViewButtons>ChangeOwner</listViewButtons>
      <lookupDialogsAdditionalFields>firstQuote__c</lookupDialogsAdditionalFields>

      <lookupDialogsAdditionalFields>finalQuote__c</lookupDialogsAdditionalFields>

      <massQuickActions>Create_MQA_Contact</massQuickActions>
      <searchResultsAdditionalFields>CREATEDBY_USER</searchResultsAdditionalFields>

    </searchLayouts>
    . . .
  </CustomObject>
```

Wildcard Support in the Manifest File

This metadata type doesn't support the wildcard character * (asterisk) in the `package.xml` manifest file. For information about using the manifest file, see [Deploying and Retrieving Metadata with the Zip File](#).

SynonymDictionary

Represents a set of synonym groups, which are groups of words or phrases that are treated as equivalent in users' searches. You can define synonym groups to optimize search results for acronyms, variations of product names, and other terminology unique to your organization.

Synonyms are available in Service Cloud features such as Salesforce Knowledge. This type extends the Metadata metadata type and inherits its `fullName` field.

File Suffix and Directory Location

SynonymDictionary components have the suffix `.synonymDictionary` and are stored in the `synonymDictionaries` folder.

Version

SynonymDictionary components are available in API version 29.0 and later.

Special Access Rules

Synonyms must be enabled in your organization. Only users with the "Manage Synonyms" permission can access this object.

Fields

Field Name	Field Type	Description
<code>groups</code>	SynonymGroup	The synonym groups defined in this dictionary.
<code>isProtected</code>	boolean	Indicates whether this component is protected (<code>true</code>) or not (<code>false</code>). Protected components cannot be linked to or referenced by components created in the installing organization.
<code>label</code>	string	Required. Specifies the display name of the synonym dictionary.

SynonymGroup

Represents a group of synonymous words or phrases.

Field Name	Field Type	Description
<code>languages</code>	Language	Required. Specifies the languages the synonym group applies to. If synonyms are specific to a single language, specify only that language. If the synonyms apply to multiple languages, specify multiple languages for one synonym group.

Field Name	Field Type	Description
terms	string	<p>Required. A word or phrase synonymous with other terms in the group. Maximum of 50 characters. Minimum of two <code>terms</code> per group.</p> <p>Synonym groups are symmetric, which means that if oranges and apples are defined in a synonym group, a search for <i>oranges</i> will return a match for <i>apples</i>, and vice versa for a search for <i>apples</i>.</p>

Declarative Metadata Sample Definition

The following is an example of a SynonymDictionary component:

```
<?xml version="1.0" encoding="UTF-8"?>
<SynonymDictionary xmlns="http://soap.sforce.com/2006/04/metadata">
  <groups>
    <languages>en_US</languages>
    <terms>Salesforce</terms>
    <terms>salesforce.com</terms>
    <terms>The Customer Company</terms>
    <terms>SFDC</terms>
  </groups>
  <groups>
    <languages>fr</languages>
    <terms>renault</terms>
    <terms>clio</terms>
  </groups>
  <label>Sample Dictionary</label>
</SynonymDictionary>
```

The following is an example `package.xml` that references the SynonymDictionary component.

```
<?xml version="1.0" encoding="UTF-8"?>
<Package xmlns="http://soap.sforce.com/2006/04/metadata">
  <types>
    <members>Sample Dictionary</members>
    <name>SynonymDictionary</name>
  </types>
  <version>55.0</version>
</Package>
```

Usage

If you have existing synonym groups defined before API version 29.0, your existing groups are associated with a default dictionary called `_Default`.


If you have a set of synonyms that require frequent updates, we recommend assigning the synonym group or groups to a dedicated dictionary with a small number of groups. Each time you deploy an existing dictionary, all of its synonym groups are overwritten. We don't support deploying updates to only a single synonym group within a dictionary.

Wildcard Support in the Manifest File

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

ExternalDataSource

Represents the metadata associated with an external data source. Create external data sources to manage connection details for integration with data and content that are stored outside your Salesforce org.

 **Note:** All credentials stored within this entity are encrypted under a framework that is consistent with other encryption frameworks on the platform. Salesforce encrypts your credentials by auto-creating org-specific keys. Credentials encrypted using the previous encryption scheme have been migrated to the new framework.

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

File Suffix and Directory Location

ExternalDataSource components are stored in the `dataSources` directory of the corresponding package directory. ExternalDataSource components have the suffix `.dataSource`, and the prefix is the name of the external data source.

Version


ExternalDataSource components are available in API version 28.0 and later.

Special Access Rules

As of Spring '20 and later, only authenticated internal and external users can access this type.

Fields

Field Name	Field Type	Description
<code>authProvider</code>	string	The authentication provider represented by the AuthProvider component.
<code>certificate</code>	string	If you specify a certificate, your Salesforce org supplies it when establishing each two-way SSL connection with the external system. The certificate is used for digital signatures, which verify that requests are coming from your Salesforce org.

 **Tip:** For best performance, verify that your remote HTTPS encrypted sites have OCSP (Online Certificate Status Protocol) stapling turned on.

Field Name	Field Type	Description
<code>customConfiguration</code>	string	<p>A string of configuration parameters that are specific to the external data source's type.</p> <ul style="list-style-type: none"> customConfiguration for Salesforce Connect—Cross-Org Adapter customConfiguration for Salesforce Connect—OData 2.0 or 4.0 Adapter customConfiguration for Salesforce Connect—Custom Adapter
<code>customHttpHeaders</code>	CustomHttpHeaders[]	Represents custom HTTP headers used with OData 2.0 or OData 4.0 connectors. Available in API version 43.0 or later.
<code>endpoint</code>	string	<p>The URL of the external system, or if that URL is defined in a named credential, the named credential URL. Corresponds to <code>URL</code> in the user interface.</p> <p>A named credential URL contains the scheme <code>callout:</code>, the name of the named credential, and an optional path. For example: <code>callout:My_Named_Credential/some_path</code>.</p> <p>You can append a query string to a named credential URL. Use a question mark (?) as the separator between the named credential URL and the query string. For example:</p> <p><code>callout:My_Named_Credential/some_path?format=json</code>.</p>
<code>externalDataSrcDescriptors</code>	ExternalDataSrcDescriptors[]	Represents schema descriptor for an external data source used with Salesforce Connect adapter for Amazon DynamoDB. Available in API version 55.0 or later.
<code>isWritable</code>	boolean	<p>Lets the Lightning Platform and users in this org create, update, and delete records for external objects associated with the external data source. The external object data is stored outside the org. By default, external objects are read only. Corresponds to <code>Writable External Objects</code> in the user interface.</p> <p>Available in API version 35.0 and later. However, with the cross-org adapter for Salesforce Connect, you can set this field to <code>true</code> only in API version 39.0 and later.</p>
<code>label</code>	string	<p>A user-friendly name for the external data source. The label is displayed in the Salesforce user interface, such as in list views.</p> <p>Examples include Acme Team Marketing Site, or Acme SharePoint.</p>
<code>namedCredential</code>	string	Represents the definition of the referenced named credential for an external data source of the type Amazon DynamoDB.
<code>oauthRefreshToken</code>	string	The OAuth refresh token. Used to obtain a new access token for an end user when a token expires.
<code>oauthScope</code>	string	Specifies the scope of permissions to request for the access token. Corresponds to <code>Scope</code> in the user interface.

Field Name	Field Type	Description
oauthToken	string	The access token issued by the external system.
password	string	The password to be used by your org to access the external system. Ensure that the credentials you use have adequate privileges to access the external system, perform searches, return data, and return information about the external system's metadata.
principalType	ExternalPrincipalType (enumeration of type string)	Determines whether you're using one set or multiple sets of credentials to access the external system. Corresponds to <code>Identity Type</code> in the user interface. The valid values are: <ul style="list-style-type: none"> Anonymous PerUser NamedUser
protocol	AuthenticationProtocol (enumeration of type string)	The authentication protocol that's required to access the external system. The valid values are: <ul style="list-style-type: none"> NoAuthentication Oauth Password For cloud-based Files Connect external systems, select Oauth 2.0 . For on-premises systems, select Password Authentication . For Simple URL data sources, select No Authentication .
repository	string	Used for SharePoint Online. If metadata is not accessible, use this field to create tables and default table fields.
type	ExternalDataSourceType (enumeration of type string)	Required. For Salesforce Connect, specifies the adapter that connects to the external system. The valid values are: <ul style="list-style-type: none"> AmazonDynamoDB—Amazon DynamoDB OData—OData 2.0 adapter OData4—OData 4.0 adapter SfdcOrg—cross-org adapter ApexClassId—<code>DataSource.Provider</code> class that defines the custom adapter created via the Apex Connector Framework For Files Connect, specifies the data source type. The valid values are: <ul style="list-style-type: none"> ContentHubSharepoint—SharePoint 2010 or 2013 ContentHubSharepointOffice365—SharePoint Online ContentHubSharepointOneDrive—OneDrive for Business ContentHubGDrive—Google Drive

Field Name	Field Type	Description
		<ul style="list-style-type: none"> <code>ContentHubIsotope</code>—Isotope <p>If Chatter is enabled, you can also specify <code>SimpleURL</code> to access data hosted on a web server that doesn't require authentication.</p> <ul style="list-style-type: none"> <code>outgoingemail</code>—A data source used for sending an email through a quick action. <p>The <code>Identity</code> and <code>Wrapper</code> types are reserved for future use. For the federated search external data source type, the valid value is <code>OpenSearch</code>.</p>
<code>username</code>	string	The username to be used by your org to access the external system. Ensure that the credentials you use have adequate privileges to access the external system, perform searches, return data, and return information about the external system's metadata.
<code>version</code>	string	Reserved for future use.

CustomHttpHeaders

Represents a custom HTTP header used with OData 2.0 or OData 4.0 connectors. Available in API version 43.0 or later.

Field Name	Field Type	Description
<code>description</code>	string	A text description of the header field's purpose.
<code>headerFieldName</code>	string	Required. Name of the header field. The name must contain at least one alphanumeric character or underscore. It can also include: <code>! # \$ % & ' * + - . ^ _ ` ~</code>
<code>headerFieldValue</code>	string	Required. A formula that resolves to the value for the header. The values in the formula must evaluate to a string. If the formula resolves to null and an empty string, the header isn't sent.
<code>isActive</code>	boolean	Indicates whether the custom HTTP header is available to use (<code>true</code>) or unavailable (<code>false</code>).

customConfiguration for Salesforce Connect—Cross-Org Adapter

The following sample JSON-encoded configuration string defines parameters that apply when the external data source's `type` is set to `SfdcOrg`.

```
{ "apiVersion": "32.0", "environment": "CUSTOM",
  "searchEnabled": "true", "timeout": "120" }
```

The parameters correspond to these fields in the user interface:

- `apiVersion`—API Version
- `environment`—Connect to

- `searchEnabled`—Enable Search
- `timeout`—Connection Timeout

customConfiguration for Salesforce Connect—OData 2.0 or 4.0 Adapter

The following JSON-encoded configuration string defines parameters that apply when the external data source's `type` is set to `OData` or `OData4`.

```
{ "inlineCountEnabled": "true", "csrfTokenName": "X-CSRF-Token",
  "requestCompression": "false", "pagination": "CLIENT",
  "noIdMapping": "false", "format": "ATOM",
  "searchFunc": "", "compatibility": "DEFAULT",
  "csrfTokenEnabled": "true", "timeout": "120",
  "searchEnabled": "true" }
```

The parameters correspond to these fields in the user interface.

- `compatibility`—Special Compatibility
- `csrfTokenEnabled`—CSRF Protection
- `csrfTokenName`—Anti-CSRF Token Name
- `format`—Format
- `inlineCountEnabled`—Request Row Counts
- `noIdMapping`—High Data Volume
- `pagination`—Server Driven Pagination
- `requestCompression`—Compress Requests
- `searchEnabled`—Enable Search
- `searchFunc`—Custom Query Option for Salesforce Search
- `timeout`—Connection Timeout

customConfiguration for Salesforce Connect—Custom Adapter

The following sample JSON-encoded configuration string defines the parameter that applies when the external data source's `type` is set to the ID of a `DataSource.Provider` class.

```
{ "noIdMapping": "false" }
```

The `noIdMapping` parameter corresponds to the `High Data Volume` field in the user interface.

Declarative Metadata Sample Definition

The following is the definition of an external data source for Salesforce Connect—OData 2.0 or 4.0 adapter.

```
<?xml version="1.0" encoding="UTF-8"?>
<ExternalDataSource xmlns="http://soap.sforce.com/2006/04/metadata">
  <authProvider>FacebookAuth</authProvider>
  <customConfiguration>{"compatibility": "DEFAULT",
    "noIdMapping": "false", "inlineCountEnabled": "true",
    "searchEnabled": "true", "format": "ATOM",
    "requestCompression": "false", "pagination": "SERVER",
```

```

    "timeout": "120"}</customConfiguration>
  <customHttpHeaders>
    <headerFieldName>X-User</headerFieldName>
    <headerFieldValue>$User.Username</headerFieldValue>
  </customHttpHeaders>
  <endpoint>http://myappname.herokuapp.com/DataHub.svc</endpoint>
  <label>DataHub</label>
  <principalType>NamedUser</principalType>
  <protocol>OAuth</protocol>
  <type>OData</type>
</ExternalDataSource>

```

ExternalDataSrcDescriptors for Salesforce Connect Adapter for Amazon DynamoDB

Represents schema descriptor for an external data source used with Salesforce Connect adapter for Amazon DynamoDB. Available in API version 55.0 or later.

Field Name	Field Type	Description
customObject	string	If set, the external object associated with the descriptor.
descriptor	string	Required. The descriptor document that contains the metadata information.
descriptorVersion	string	If the external system supports schema versioning for the data source, the optional descriptor document version tracks the external system's schema version. Several descriptors with different document versions may be active.
developerName	string	Required. The unique name of the child-level setup entity.
externalDataSource	string	Required. The name of the external data source associated with the descriptor.
subtype	ExternalDataSrcDescType (enumeration of type string)	Required. The subtype of the descriptor. Values are: <ul style="list-style-type: none"> SchemaTableMetadata— Used to cache information about the external system. SchemaTableQualifiers— Used to customize the data retrieval query to the external system.
systemVersion	int	Required. The version that defines the descriptor format and allows compatibility with descriptor formats between Salesforce releases.
type	ExternalDataSrcDescType (enumeration of type string)	Required. The type of the descriptor. Valid value: <ul style="list-style-type: none"> Schema

Declarative Metadata Sample Definition

The following is an example of an external data source for the Salesforce Connect adapter for Amazon DynamoDB that uses `ExternalDataSrcDescriptor` component.

```
<?xml version="1.0" encoding="UTF-8"?>
<ExternalDataSource xmlns="http://soap.sforce.com/2006/04/metadata">
  <customConfiguration>{"timeout":"120"}</customConfiguration>
  <externalDataSrcDescriptors>
    <fullName>MyQualifierName</fullName>
    <customObject>MyExternalObject__x</customObject>
    <descriptor>
      {
        "tableName": "MyDynamoDBTable",
        "columns": {
          "MyField": {"presence": "exists"}
        }
      }
    </descriptor>
    <developerName>MyQualifierName</developerName>
    <externalDataSource>MyDataSource</externalDataSource>
    <subtype>SchemaTableQualifiers</subtype>
    <systemVersion>0</systemVersion>
    <type>Schema</type>
  </externalDataSrcDescriptors>
  <isWritable>true</isWritable>
  <label>MyDataSource</label>
  <namedCredential>MyNamedCredential</namedCredential>
  <principalType>Anonymous</principalType>
  <protocol>NoAuthentication</protocol>
  <type>AmazonDynamoDb</type>
</ExternalDataSource>
```

Wildcard Support in the Manifest File

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

CHAPTER 5 Salesforce Knowledge with SOQL and SOSL

In this chapter ...

- Update an Article’s Keyword Tracking with SOQL
- Update an Article Viewstat with SOQL
- WITH filteringExpression
- Using Relationship Queries with Data Category Selection Objects
- About SOSL

Use the Salesforce Object Query Language (SOQL) to search your organization’s Knowledge data for specific information. SOQL is similar to the SELECT statement in the widely used Structured Query Language (SQL) but is designed specifically for Salesforce. Use the Salesforce Object Search Language (SOSL) to construct text-based search queries against the search index.

Knowledge Object	Limits
KnowledgeArticleVersion	<ul style="list-style-type: none">• Filter on a single value of <code>PublishStatus</code> for best results. To find all versions of each article, omit the <code>PublishStatus</code> filter, but do filter on one or more primary key IDs. To retrieve all archived versions for a given article, specify a SOQL filter where <code>IsLatestVersion</code> is <code>false</code>.• In API version 46.0 and earlier, queries without a filter on <code>PublishStatus</code> return published articles by default. In API version 47.0 and later, draft, published, and archived articles are returned when Lightning Knowledge is enabled.• To support security, only users with the “View Draft Articles” permission see articles whose <code>PublishStatus</code> value is <code>Draft</code>. Similarly, only users with the “View Archived Articles” permission see articles whose <code>PublishStatus</code> value is <code>Archived</code>.• Archived article versions are stored in the Knowledge__kav object. To query archived article versions, specify the article <code>Id</code> and set <code>IsLatestVersion='0'</code>.• You can’t use binding variables in Apex SOQL statements with <code>KnowledgeArticleVersion</code> objects. For example, the following SOQL statement causes a compilation error. <pre>final String PUBLISH_STATUS_ONLINE = 'Online'; List<Knowledge__kav> articles</pre>

Knowledge Object	Limits
	<pre data-bbox="1013 268 1442 457">= [SELECT Id FROM Knowledge__kav WHERE PublishStatus = :PUBLISH_STATUS_ONLINE];</pre> <p data-bbox="1013 478 1419 541">Instead, use dynamic SOQL as follows. See Dynamic SOQL in <i>Apex Developer Guide</i>.</p> <pre data-bbox="1013 562 1442 877">final String PUBLISH_STATUS_ONLINE = 'Online'; final String q = 'SELECT Id, PublishStatus FROM Knowledge__kav WHERE PublishStatus = :PUBLISH_STATUS_ONLINE'; List<Knowledge__kav> articles = Database.query(q);</pre>

For generic SOQL and SOSL information see the [SOQL and SOSL Reference](#).

Update an Article's Keyword Tracking with SOQL

Track keywords that are used in Salesforce Knowledge article searches with the `UPDATE TRACKING` optional clause on a SOQL query. `UPDATE TRACKING` is an optional clause that can be added to a `SELECT` statement of a SOQL query to report on article searches and views. Developers can use `UPDATE TRACKING` to track the keywords that are used in Salesforce Knowledge article searches.



Example: You can use this syntax to track a keyword that are used in Salesforce Knowledge article search:

```
SELECT Title FROM FAQ__kav
WHERE Keyword='Apex' and
Language = 'en_US' and
KnowledgeArticleVersion = 'ka230000000PCiy'
UPDATE TRACKING
```

Update an Article Viewstat with SOQL

Determine how many hits a Salesforce Knowledge article has had by using the `UPDATE VIEWSTAT` optional clause on a SOQL query. You can get a view count for every article that you have access to online.

The `UPDATE VIEWSTAT` clause is used in a `SELECT` statement to report on Salesforce Knowledge article searches and views. It allows developers to update an article's view statistics.

You can use this syntax to increase the view count for every article you have access to online:

```
SELECT Title FROM Knowledge__kav
WHERE PublishStatus='online' and
Language = 'en_US' and
KnowledgeArticleVersion = 'ka230000000PCiy'
UPDATE VIEWSTAT
```

WITH *filteringExpression*

You can filter records based on field values, for example, to filter according to category or to query and retrieve changes that are tracked in a user's profile feed by using `WITH filteringExpression`. This optional clause can be added to a `SELECT` statement of a SOQL query.

Unlike the `WHERE` clause, which only supports fields from the object specified in the `FROM` clause, `WITH` allows you to filter by other related criteria. For example, you can use the `WITH` clause to filter articles based on their classification in one or more data category groups. The `WITH` clause can only be used in the following cases:

- To filter records based on their categorization. See [WITH DATA CATEGORY filteringExpression](#).
- To query and retrieve record changes tracked in a user profile feed. See Custom Object__Feed in the [Object Reference for Salesforce](#).
- To filter records based on user access permissions on fields or objects being queried within Apex code. See [Filter SOQL Queries Using WITH SECURITY_ENFORCED](#) in the *Apex Developer Guide*.
- To specify user mode access for Apex database operations. Apex code runs in system mode by default, which means that it runs with substantially elevated permissions over the user running the code. To enhance the security context of Apex, you can specify user mode access. Field-level security (FLS) and object permissions of the running user are respected in user mode, unlike in system mode. User mode always applies sharing rules; in system mode they're controlled by class sharing keywords. See [Enforce User Mode for Database Operations \(Beta\)](#) in the *Apex Developer Guide*.

If **WITH** is specified, the query returns only records that match the filter and are visible to the user. If unspecified, the query returns only the matching records that are visible to the user.

The filtering expression in this statement is highlighted in bold. The syntax is explained in the following sections.

- `SELECT Title FROM KnowledgeArticleVersion WHERE PublishStatus='online' WITH DATA CATEGORY Geography__c ABOVE usa__c`
- `SELECT Id FROM UserProfileFeed WITH UserId='005D000001AamR' ORDER BY CreatedDate DESC, Id DESC LIMIT 20`

Using RecordVisibilityContext

You can also use `RecordVisibilityContext` to filter `WITH` clauses. This feature is available in API version 48.0 and later.

For example:

```
SELECT Id, RecordVisibility.VisibilityAttribute FROM Account WHERE Id = 'xxx'
  WITH RecordVisibilityContext (maxDescriptorPerRecord=100, supportsDomains=true,
  supportsDelegates=true)
```

Or:

```
SELECT recordId, VisibilityAttribute FROM RecordVisibility WHERE recordId = 'xxx'
  WITH RecordVisibilityContext (maxDescriptorPerRecord=100, supportsDomains=true,
  supportsDelegates=true)
```

The following fields can be used as parameters:

Field	Description	Default
<code>maxDescriptorPerRecord</code>	The maximum number of descriptors to return per record. If the actual number of descriptors for a specific record exceeds that value, then a "tooManyDescriptors" visibility attribute is returned for that record.	<ul style="list-style-type: none"> • orgs with CRM Analytics: 150 • other orgs: 400
<code>supportsDomains</code>	Whether the Visibility Service is allowed to generate "domain" visibility attributes.	true
<code>supportsDelegates</code>	Whether the Visibility Service is allowed to generate "delegate" visibility attributes.	true

At least one field is necessary. This query is *invalid*:

```
SELECT Id, RecordVisibility.VisibilityAttribute FROM Account WHERE Id = 'xxx'
  WITH RecordVisibilityContext
```

This query is valid:

```
SELECT Id, RecordVisibility.VisibilityAttribute FROM Account WHERE Id = 'xxx'
  WITH RecordVisibilityContext (maxDescriptorPerRecord=100)
```

For more information, see [RecordVisibility](#) in the *Object Reference*.

IN THIS SECTION:


[WITH DATA CATEGORY filteringExpression](#)

You can search for Salesforce Knowledge articles and questions by their data category in a SOQL query. `WITH DATA CATEGORY` is an optional clause in a `SELECT` statement that's used to filter records that are associated with one or more data categories and are visible to users.

WITH DATA CATEGORY *filteringExpression*

You can search for Salesforce Knowledge articles and questions by their data category in a SOQL query. `WITH DATA CATEGORY` is an optional clause in a `SELECT` statement that's used to filter records that are associated with one or more data categories and are visible to users.

If `WITH DATA CATEGORY` is specified, the `query()` returns only matching records that are associated with the specified data categories and are visible to the user. If unspecified, the `query()` only returns the matching records that are visible to the user.

 **Important:** `CategoryData` is an object and `DATA CATEGORY` is syntax in a SOQL `WITH` clause. `WITH DATA CATEGORY` is valid syntax, but `WITH CategoryData` is not supported.


A SOQL statement using a `WITH DATA CATEGORY` clause must also include a `FROM ObjectTypeName` clause where `ObjectTypeName` equals:

- `KnowledgeArticleVersion` to query all article types
- an article type API Name to query a specific article type
- `Question` to query questions

When `ObjectTypeName` equals to `KnowledgeArticleVersion` or any article type API Name in the `FROM` clause, a `WHERE` clause must be specified with one of the following parameters:

- `PublishStatus` to query articles depending on their status in the publishing cycle:
 - `WHERE PublishStatus='online'` for published articles
 - `WHERE PublishStatus='archived'` for archived articles
 - `WHERE PublishStatus='draft'` for draft articles
- `Id` to query an article based on its id

For information on article types or questions, see “Knowledge Article Types” in the Salesforce Help.

 **Note:** The `WITH DATA CATEGORY` clause does not support bind variables. The clause can have no more than three data category conditions.

filteringExpression

The *filteringExpression* in the `WITH DATA CATEGORY` clause uses the following syntax:

```
dataCategorySelection [AND [dataCategorySelection] [...]]
```

The examples in this section are based on the following data category group:

```
Geography__c
  ww__c
    northAmerica__c
      usa__c
      canada__c
```



```

    mexico__c
  europe__c
    france__c
    uk__c
  asia__c

```

The category filtering in the statements below is highlighted in bold. The syntax is explained in the following sections.

- **SELECT Title FROM KnowledgeArticleVersion WHERE PublishStatus='online' WITH DATA CATEGORY Geography__c ABOVE usa__c**
- **SELECT Title FROM Question WHERE LastReplyDate > 2005-10-08T01:02:03Z WITH DATA CATEGORY Geography__c AT (usa__c, uk__c)**
- **SELECT UrlName FROM KnowledgeArticleVersion WHERE PublishStatus='draft' WITH DATA CATEGORY Geography__c AT usa__c AND Product__c ABOVE_OR_BELOW mobile_phones__c**

You can only use the AND logical operator. The following syntax is incorrect as OR is not supported:

```
WITH DATA CATEGORY Geography__c ABOVE usa__c OR Product__c AT mobile_phones__c
```

IN THIS SECTION:

[dataCategorySelection](#)

The syntax of the data category selection in a WITH DATA CATEGORY clause in a SOQL query includes a category group name to use as a filter, the filter selector, and the name of the category to use for filtering.

[Filtering Selectors](#)

When specifying filters for a WITH CATEGORY clause of a SOQL query, you can use AT to select the specified category, ABOVE to select the category and all its parent categories, BELOW to select the category and all its subcategories, and ABOVE_OR_BELOW to select the category, its parent categories, and its subcategories.

[Example WITH DATA CATEGORY Clauses](#)

Here are examples of WITH DATA CATEGORY clauses in a SELECT statement in a SOQL query.

dataCategorySelection

The syntax of the data category selection in a WITH DATA CATEGORY clause in a SOQL query includes a category group name to use as a filter, the filter selector, and the name of the category to use for filtering.

The *dataCategorySelection* uses the following syntax:

```
dataCategoryGroupName filteringSelector dataCategoryName
```

Syntax	Description
<i>dataCategoryGroupName</i>	The name of the data category group to use as a filter. Geography__c is the data category group in the following example. You cannot use the same data category group more than once in a query. As an example, the following command is incorrect: WITH DATA CATEGORY Geography__c ABOVE usa__c AND Geography__c BELOW europe__c
<i>filteringSelector</i>	The selector used to filter the data in the specified data category. See Filtering Selectors for a list of valid selectors.

Syntax	Description
<i>dataCategoryName</i>	<p>The name of the data category for filtering. You must have visibility on the category you specify. For more information on category visibility, see "Data Category Visibility" in the Salesforce Help.</p> <p>You can use parentheses to apply the filtering operator to more than one data category. Each data category must be separated by a comma.</p> <p>Example: WITH DATA CATEGORY Geography__c AT (usa__c,france__c,uk__c)</p> <p>You can't use the AND operator instead of parentheses to list multiple data categories. The following syntax does not work WITH DATA CATEGORY Geography__c AT usa__c AND france__c</p>

Filtering Selectors

When specifying filters for a WITH CATEGORY clause of a SOQL query, you can use AT to select the specified category, ABOVE to select the category and all its parent categories, BELOW to select the category and all its subcategories, and ABOVE_OR_BELOW to select the category, its parent categories, and its subcategories.

The following table lists the *filteringSelector* values that are used in the *dataCategorySelection* syntax.

The examples in this section are based on the following data category group:


```

Geography__c
  ww__c
    northAmerica__c
      usa__c
      canada__c
      mexico__c
    europe__c
      france__c
      uk__c
    asia__c

```

Selector	Description
AT	<p>Select the specified data category.</p> <p>For example, the following syntax selects <code>asia__c</code>.</p> <pre>WITH DATA CATEGORY Geography__c AT asia__c</pre>
ABOVE	<p>Select the specified data category and all its parent categories.</p> <p>For example, the following syntax selects <code>usa__c</code>, <code>northAmerica__c</code>, and <code>ww__c</code>.</p> <pre>WITH DATA CATEGORY Geography__c ABOVE usa__c</pre>
BELOW	<p>Select the specified data category and all its subcategories.</p> <p>For example the following selects <code>northAmerica__c</code>, <code>usa__c</code>, <code>canada__c</code>, and <code>mexico__c</code>.</p> <pre>WITH DATA CATEGORY Geography__c BELOW northAmerica__c</pre>

Selector	Description
ABOVE_OR_BELOW	<p>Select the specified data category and:</p> <ul style="list-style-type: none"> all its parent categories all its subcategories <p>For example the following selects <code>ww__c</code>, <code>europa__c</code>, <code>france__c</code> and <code>uk__c</code>.</p> <pre>WITH DATA CATEGORY Geography__c ABOVE_OR_BELOW europa__c</pre>

 **Note:** For more information on data category groups, data categories, parent and subcategories, see "Work with Data Categories" in the Salesforce Help.

Example WITH DATA CATEGORY Clauses

Here are examples of `WITH DATA CATEGORY` clauses in a `SELECT` statement in a SOQL query.

Type of Search	Example(s)
Select the title from all questions classified with the <code>mobile_phones__c</code> data category in the <code>Product__c</code> data category group	<pre>SELECT Title FROM Question WHERE LastReplyDate < 2005-10-08T01:02:03Z WITH DATA CATEGORY Product__c AT mobile_phones__c</pre>
Select the title and summary from all published Knowledge articles classified: <ul style="list-style-type: none"> above or below <code>europa__c</code> in the <code>Geography__c</code> data category group below <code>allProducts__c</code> in the <code>Product__c</code> data category group 	<pre>SELECT Title, Summary FROM KnowledgeArticleVersion WHERE PublishStatus='Online' AND Language = 'en_US' WITH DATA CATEGORY Geography__c ABOVE_OR_BELOW europa__c AND Product__c BELOW All__c</pre>
Select the ID and title from draft articles of type "Offer__kav" classified : <ul style="list-style-type: none"> with the <code>france__c</code> or <code>usa__c</code> data category in the <code>Geography__c</code> data category group above the <code>ds1__c</code> data category in the <code>Product__c</code> data category group 	<pre>SELECT Id, Title FROM Offer__kav WHERE PublishStatus='Draft' AND Language = 'en_US' WITH DATA CATEGORY Geography__c AT (france__c,usa__c) AND Product__c ABOVE ds1__c</pre>

Using Relationship Queries with Data Category Selection Objects

Data categories are used to classify records. In SOQL, you can use the `Article__DataCategorySelection` or `QuestionDataCategorySelection` objects. You can also build a relationship query with the `DataCategorySelections` relationship name in a `FROM` clause.

Imagine an `Offer` article type. The following query returns the ID of any categorization associated with an offer and the ID of the categorized article.

```
SELECT Id, ParentId
FROM Offer__DataCategorySelection
```

The following example uses the `DataCategorySelections` relationship name to build a relationship query that returns the ID of published offers and the ID of all the categorizations associated to these offers.

```
SELECT Id, Title,
(
  SELECT Id
  FROM DataCategorySelections
)
FROM Offer__kav WHERE PublishStatus='online'
```

About SOSL

Use the Salesforce Object Search Language (SOSL) to construct text-based search queries against the search index.

You can search text, email, and phone fields for multiple objects, including custom objects, that you have access to in a single query in the following environments.

- SOAP or REST calls
- Apex statements
- Visualforce controllers and getter methods
- Schema Explorer of the Eclipse Toolkit

 **Note:** If your org has relationship queries enabled, SOSL supports SOQL relationship queries.

Compare SOSL and SOQL

Use SOQL when:

- You want to search against the org's database. Results from a database search include matches for the exact search string.
- You know in which objects or fields the data resides.
- You want to:
 - Retrieve data from a single object or from multiple objects that are related to one another
 - Count the number of records that meet specified criteria
 - Sort results as part of the query
 - Retrieve data from number, date, or checkbox fields

Use SOSL when:

- You don't know in which object or field the data resides, and you want to find it in the most efficient and fastest way possible.
- You want to:
 - Retrieve data for a specific term that you know exists within a field. Because SOSL can tokenize multiple terms within a field and build a search index off this, SOSL searches are faster and can return more relevant results. Depending on the search object, SOSL searches also take advantage of the advanced features of the search index, such as out-of-order matching, synonyms, lemmatization, and spell check.

- Retrieve multiple objects and fields efficiently, and the objects might or might not be related to one another.
- Retrieve data for a particular division in an org using the divisions feature, and you want to find it in the most efficient way possible.

Define Efficient SOSL Text Searches

If your searches are too general, they are slow and return too many results. Use the following clauses to write more targeted and useful searches.

- `IN`: Limits the types of fields to search, including email, name, or phone.
- `LIMIT`: Specifies the maximum number of rows to return.
- `OFFSET`: Displays the search results on multiple pages.
- `RETURNING`: Limits the objects and fields to return.
- `WITH DATA CATEGORY`: Specifies the data categories to return.
- `WITH DivisionFilter`: Specifies the division field to return.
- `WITH NETWORK`: Specifies the Experience Cloud site ID to return.
- `WITH PricebookId`: Specifies the price book ID to return.

IN THIS SECTION:

[Update an Article's Keyword Tracking with SOSL](#)

Track keywords that are used in Salesforce Knowledge article searches with the `UPDATE TRACKING` optional clause on a SOSL query. You can use the language attribute to search by locale.

[Update an Article's Viewstat with SOSL](#)

Determine how many hits a Salesforce Knowledge article has had by using the `UPDATE VIEWSTAT` optional clause on a SOSL query. You can use the language attribute to search by locale.

[WITH DATA CATEGORY DataCategorySpec](#)

`WITH DATA CATEGORY` is an optional clause that can be added to a SOSL query to filter all search results that are associated with one or more data categories and are visible to users. This clause is used in searches of Salesforce Knowledge articles and questions.

[WITH SNIPPET](#)

`WITH SNIPPET` is an optional clause that can be added to a SOSL query for article, case, feed, and idea searches. On the search results page, excerpts below article titles show terms matching the search query highlighted within the context of surrounding text. Snippets make it easier for users to identify the content they're looking for.

Update an Article's Keyword Tracking with SOSL

Track keywords that are used in Salesforce Knowledge article searches with the `UPDATE TRACKING` optional clause on a SOSL query. You can use the language attribute to search by locale.

The `UPDATE TRACKING` clause is used to report on Salesforce Knowledge article searches and views. It allows developers to track the keywords used in Salesforce Knowledge article searches. Also, the language attribute can be used to search by a specific language (locale). However, only one language can be specified in a single query. Make a separate query for each language that you want. Use the Java format, which uses the underscore (for example, `fr_FR`, `jp_JP`, and so on), to supply locales. Search the Web for "java locale codes" to get a list of supported locales.

You can use this syntax to track a keyword used in Salesforce Knowledge article search:

```
FIND {Keyword}
RETURNING KnowledgeArticleVersion (Title WHERE PublishStatus="Online" and language="en_US")
UPDATE TRACKING
```

Update an Article's Viewstat with SOSL

Determine how many hits a Salesforce Knowledge article has had by using the `UPDATE VIEWSTAT` optional clause on a SOSL query. You can use the `language` attribute to search by locale.

The optional `UPDATE VIEWSTAT` clause is used to report on Salesforce Knowledge article searches and views. It allows developers to update an article's view statistics. Also, the `language` attribute can be used to search by a specific language (locale). However, only one language can be specified in a single query. Make a separate query for each language that you want. Use the Java format, which uses the underscore (for example, `fr_FR`, `jp_JP`, and so on), to supply locales. Search the Web for "java locale codes" to get a list of supported locales.

You can use this syntax to increase the view count for every article you have access to online in US English:

```
FIND {Title}
RETURNING FAQ__kav (Title WHERE PublishStatus="Online" and
language="en_US" and
KnowledgeArticleVersion = 'ka230000000PCiy')
UPDATE VIEWSTAT
```

WITH DATA CATEGORY *DataCategorySpec*

`WITH DATA CATEGORY` is an optional clause that can be added to a SOSL query to filter all search results that are associated with one or more data categories and are visible to users. This clause is used in searches of Salesforce Knowledge articles and questions.

The `WITH DATA CATEGORY` clause can be used in API version 18.0 or later.

Syntax

The `WITH DATA CATEGORY` syntax is:

```
WITH DATA CATEGORY DataCategorySpec [logicalOperator DataCategorySpec2 ... ]
```

Where *DataCategorySpec* consists of a *groupName*, *Operator*, and *category*.

Name	Description
<i>groupName</i>	The name of the data category group to filter. For information on category groups, see "Create and Modify Category Groups" in the Salesforce Help.
<i>Operator</i>	Use one of the following operators: <ul style="list-style-type: none"> • <code>AT</code>—Queries the specified data category. • <code>ABOVE</code>—Queries the specified data category and all of its parent categories. • <code>BELOW</code>—Queries the specified data category and all of its subcategories. • <code>ABOVE_OR_BELOW</code>—Queries the specified data category, all of its parent categories, and all of its subcategories.

Name	Description
<code>category</code>	The name of the category to filter. To include multiple data categories, enclose them in parentheses, separated by commas. For information on categories, see “Add Data Categories to Category Groups” in the Salesforce Help.

You can add multiple data category specifiers by using the logical operator `AND`. Other operators, such as `OR` and `AND NOT`, are not supported.

A SOSL statement using the `WITH DATA CATEGORY` clause must also include a `RETURNING ObjectName` clause, with a `WHERE` clause that filters on the `PublishStatus` field.

In the `RETURNING` clause, specify one of the following for `ObjectName`:

- To search a specific article type, use the article type name with the suffix `__kav`
- To search all article types, use [KnowledgeArticleVersion](#)
- To search questions, use [Question](#)

For information on article types, see “Knowledge Article Types” in the Salesforce Help.


The `WHERE` clause must use one of the following publish statuses:

- `WHERE PublishStatus='online'` for published articles
- `WHERE PublishStatus='archived'` for archived articles
- `WHERE PublishStatus='draft'` for draft articles

Examples

Search Type	Example
Search all published (online)Salesforce Knowledge articles with a category from one category group.	
Search online FAQ articles with categories from two category groups.	
Search archived FAQ articles from one category group.	
Search all draft Salesforce Knowledge articles from one category group.	

For information on the `WITH DATA CATEGORY` clause, see the [WITH DATA CATEGORY *filteringExpression*](#).

 **Tip:** You can also search for articles by ID, without using the `WITH DATA CATEGORY` clause. For more information, see [Example WHERE Clauses](#).

WITH SNIPPET

`WITH SNIPPET` is an optional clause that can be added to a SOSL query for article, case, feed, and idea searches. On the search results page, excerpts below article titles show terms matching the search query highlighted within the context of surrounding text. Snippets make it easier for users to identify the content they're looking for.

 **Note:** To generate search results with highlighted matches but not snippets, use `WITH HIGHLIGHT`.

Search snippets and highlights are generated from the following field types.

- Email
- Text
- Text Area
- Text Area (Long)
- Text Area (Rich)

Search snippets and highlights are *not* generated from the following field types.

- Checkbox
- Currency
- Date
- Date/Time
- File
- Formula
- Lookup Relationship
- Number
- Percent
- Phone
- Picklist
- Picklist (Multi-Select)
- URL

 **Example:** The following SOSL statement returns snippets for articles that match the search term *San Francisco*.

```
FIND {San Francisco} IN ALL FIELDS RETURNING KnowledgeArticleVersion(id, title WHERE
PublishStatus = 'Online' AND Language = 'en_US') WITH
SNIPPET (target_length=120)
```

The matching terms are highlighted with `<mark>` tags within the context of the snippet results. Stemmed forms of the term and any synonyms defined are also highlighted.

 **Example:**


```
[ {
  "attributes" : {
    "type" : "KnowledgeArticleVersion",
    "url" : "/services/data/v32.0/subjects/KnowledgeArticleVersion/kaKD0000000001MAA"
```



```

    },
    "Id" : "kaKD00000000001MAA"
    "Title" : "San Francisco"
    "Summary" : "City and County of San Francisco"
    "snippet.text" : "<mark>San</mark> <mark>Francisco</mark>, officially the City and
    County of <mark>San</mark> <mark>Francisco</mark> is the... City and County of
    <mark>San</mark> <mark>Fran</mark>"
    "highlight.Title" : "<mark>San</mark> <mark>Francisco</mark>"
  }, {
    "attributes" : {
      "type" : "KnowledgeArticleVersion",
      "url" : "/services/data/v32.0/subjects/KnowledgeArticleVersion/kaBD0000000007DMAQ"
    },
    "Id" : "kaBD0000000007DMAQ",
    "Title" : "San Francisco Bay Area",
    "Summary" : "Nine county metropolitan area",
    "snippet.text" : "The <mark>SF</mark> Bay Area, commonly known as the Bay Area, is
    a populated region that"
    "highlight.Title" : "<mark>San</mark> <mark>Francisco</mark> Bay Area"
  }, {
    "attributes" : {
      "type" : "KnowledgeArticleVersion",
      "url" : "/services/data/v32.0/subjects/KnowledgeArticleVersion/ka3D0000000042OIAQ"
    },
    "Id" : "ka3D0000000042OIAQ",
    "Title" : "California",
    "Summary" : "State of California",
    "snippet.text" : "(Greater Los Angeles area and <mark>San</mark>
    <mark>Francisco</mark> Bay Area, respectively), and eight of the nation's 50 most"
  } ]

```

 **Note:** In this example, “SF” (as a synonym defined for “San Francisco”) and “San Fran” (as a stemmed form of “San Francisco”) are also highlighted in the results as matching terms.

Usage

For SOSL statements using the `WITH SNIPPET` clause, we recommend using a `RETURNING ObjectTypeName` clause, with a `WHERE` clause that filters on the `PublishStatus` field.

In the `RETURNING` clause, specify one of the following for `ObjectTypeName`:

- To search a specific article type, use the article type name with the suffix `__kav`.
- To search all article types, use `KnowledgeArticleVersion`.
- To search case, case comment, feed, feed comment, idea, and idea comment types, use `Case`, `CaseComment`, `FeedItem`, `FeedComment`, `Idea`, and `IdeaComment`. For example:

```


FIND {San Francisco} IN ALL FIELDS RETURNING FeedItem, FeedComment WITH SNIPPET
(target_length=120)

```

Other objects that are included in searches that contain `WITH SNIPPET` don't return snippets.


Snippets aren't displayed for search terms that contain a wildcard, when the search doesn't return any articles, or if the user doesn't have access to the field that contains the snippet. Even if you add the `WITH SNIPPET` clause, searches that don't return snippets don't return snippets.

Snippets are only displayed when 20 or fewer results are returned on a page.

 **Tip:** Use the `LIMIT` or `OFFSET` clause to return only 20 results at a time.

Escaped HTML Tags

When matching terms within HTML tags are returned in a snippet, the HTML tags are escaped and the matching terms are highlighted in the results.

 **Example:** A search for `salesforce` returns an article with the text "For more information, visit `salesforce.com`". The original hyperlink tags from the article are escaped (encoded) and "salesforce" is highlighted in the snippet result.


```
For more information, visit &lt;a
href='https://salesforce.com'&gt;salesforce.com&lt;/a&gt;
```

Target Snippet Length

By default, each snippet displays up to approximately 300 characters, which is usually three lines of text in a standard browser window display. The number of characters displayed is the *target length*, within a statistically insignificant degree of variance.

Snippets consist of one or more *fragments* of text that contain the matching terms. If the returned snippet includes multiple text fragments (for example, for matches within multiple fields), the target length is the maximum total length of all the returned fragments.

To specify an alternate target length, add the optional `target_length` parameter to the `WITH SNIPPET` clause. You can specify a target length from 50 to 1,000 characters. When the `target_length` is set to an invalid number, such as `0` or a negative number, the length defaults to `300`.

 **Example:** A `target_length` parameter of 120 characters is useful for displaying a snippet of approximately three lines of text in a standard mobile interface.

```
FIND {San Francisco} IN ALL FIELDS RETURNING KnowledgeArticleVersion(id, title WHERE
PublishStatus = 'Online' AND Language = 'en_US') WITH
SNIPPET(target_length=120)
```

Supported APIs

The `WITH SNIPPET` clause can be used in API version 32.0 or later. The `WITH SNIPPET` clause in SOSL is supported in SOAP API, REST API, and Apex.

CHAPTER 6 Salesforce Knowledge, Apex, and Visualforce

In this chapter ...

- [PublishingService Class](#)

With the power of Apex and Visualforce, you can customize the look and feel of your knowledge base; for example, with custom search pages or a robust article publishing cycle.

Apex is a strongly typed, object-oriented programming language that allows developers to execute flow and transaction control statements on the server in conjunction with calls to the API. Using syntax that looks like Java and acts like database stored procedures, Apex enables developers to add business logic to most system events, including button clicks, related record updates, and Visualforce pages. Apex code can be initiated by Web service requests and from triggers on objects.

Visualforce is a framework that allows developers to build sophisticated, custom user interfaces that can be hosted natively on the Lightning Platform. The Visualforce framework includes a tag-based markup language, similar to HTML, and a set of server-side “standard controllers” that makes basic database operations, such as queries and saves, very simple to perform.

Due to the endless possibilities with Apex and Visualforce, only the Apex Publishing Service class is in this guide. For more Apex and Visualforce information see the [Apex Developer Guide](#) and the [Visualforce Developer Guide](#).

PublishingService Class

Use the methods in the `KbManagement.PublishingService` class to manage the lifecycle of an article and its translations.


Namespace

`KbManagement`

Usage

Use the methods in the `KbManagement.PublishingService` class to manage the following parts of the lifecycle of an article and its translations:

- Publishing
- Updating
- Retrieving
- Deleting
- Submitting for translation
- Setting a translation to complete or incomplete status
- Archiving
- Assigning review tasks for draft articles or translations

 **Note:** Date values are based on GMT.

To use the methods in this class, you must enable Salesforce Knowledge. See [Salesforce Knowledge Implementation Guide](#) for more information on setting up Salesforce Knowledge.

IN THIS SECTION:

[PublishingService Methods](#)

PublishingService Methods

The following are methods for `PublishingService`. All methods are static.

IN THIS SECTION:

[archiveOnlineArticle\(articleId, scheduledDate\)](#)

Archives an online version of an article. If the specified `scheduledDate` is null, the article is archived immediately. Otherwise, it archives the article on the scheduled date.

[assignDraftArticleTask\(articleId, assigneeId, instructions, dueDate, sendEmailNotification\)](#)

Assigns a review task related to a draft article.

[assignDraftTranslationTask\(articleVersionId, assigneeId, instructions, dueDate, sendEmailNotification\)](#)

Assigns a review task related to a draft translation.

[cancelScheduledArchivingOfArticle\(articleId\)](#)

Cancels the scheduled archiving of an online article.

[cancelScheduledPublicationOfArticle\(articleId\)](#)

Cancels the scheduled publication of a draft article.

[completeTranslation\(articleVersionId\)](#)

Puts a translation in a completed state that is ready to publish.

[deleteArchivedArticle\(articleId\)](#)

Deletes an archived article.

[deleteArchivedArticleVersion\(articleId, versionNumber\)](#)

Deletes a specific version of an archived article.

[deleteDraftArticle\(articleId\)](#)

Deletes a draft article.

[deleteDraftTranslation\(articleVersionId\)](#)

Deletes a draft translation.

[editArchivedArticle\(articleId\)](#)

Creates a draft article from the archived master version and returns the new draft master version ID of the article.

[editOnlineArticle\(articleId, unPublish\)](#)

Creates a draft article from the online version and returns the new draft master version ID of the article. Also, unpublishes the online article, if *unPublish* is set to `true`.

[editPublishedTranslation\(articleId, language, unPublish\)](#)

Creates a draft version of the online translation for a specific language and returns the new draft master version ID of the article. Also, unpublishes the article, if set to `true`.

[publishArticle\(articleId, flagAsNew\)](#)

Publishes an article. If *flagAsNew* is set to `true`, the article is published as a major version.

[restoreOldVersion\(articleId, versionNumber\)](#)

Creates a draft article from an existing online article based on the specified archived version of the article and returns the article version ID.

[scheduleForPublication\(articleId, scheduledDate\)](#)

Schedules the article for publication as a major version. If the specified date is null, the article is published immediately.

[setTranslationToIncomplete\(articleVersionId\)](#)

Sets a draft translation that is ready for publication back to “in progress” status.

[submitForTranslation\(articleId, language, assigneeId, dueDate\)](#)

Submits an article for translation to the specified language. Also assigns the specified user and due date to the submittal and returns new ID of the draft translation.

archiveOnlineArticle(articleId, scheduledDate)

Archives an online version of an article. If the specified *scheduledDate* is null, the article is archived immediately. Otherwise, it archives the article on the scheduled date.

Signature

```
public static Void archiveOnlineArticle(String articleId, Datetime scheduledDate)
```

Parameters

articleId

Type: String

scheduledDate

Type: Datetime

Return Value

Type: Void

Example

```
String articleId = 'Insert article ID';
Datetime scheduledDate = Datetime.newInstanceGmt(2012, 12, 1, 13, 30, 0);
KbManagement.PublishingService.archiveOnlineArticle(articleId, scheduledDate);
```

assignDraftArticleTask(articleId, assigneeId, instructions, dueDate, sendEmailNotification)

Assigns a review task related to a draft article.

Signature

```
public static Void assignDraftArticleTask(String articleId, String assigneeId, String instructions, Datetime dueDate, Boolean sendEmailNotification)
```

Parameters

articleId

Type: String

assigneeId

Type: String

instructions

Type: String

dueDate

Type: Datetime

sendEmailNotification

Type: Boolean

Return Value

Type: Void

Example

```
String articleId = 'Insert article ID';
String assigneeId = '';
String instructions = 'Please review this draft.';
Datetime dueDate = Datetime.newInstanceGmt(2012, 12, 1);
KbManagement.PublishingService.assignDraftArticleTask(articleId, assigneeId, instructions,
    dueDate, true);
```

assignDraftTranslationTask(articleVersionId, assigneeId, instructions, dueDate, sendEmailNotification)

Assigns a review task related to a draft translation.

Signature

```
public static Void assignDraftTranslationTask(String articleVersionId, String assigneeId,
String instructions, Datetime dueDate, Boolean sendEmailNotification)
```

Parameters

articleVersionId

Type: String

assigneeId

Type: String

instructions

Type: String

dueDate

Type: Datetime

sendEmailNotification

Type: Boolean

Return Value

Type: Void

Example

```
String articleId = 'Insert article ID';
String assigneeId = 'Insert assignee ID';
String instructions = 'Please review this draft.';
Datetime dueDate = Datetime.newInstanceGmt(2012, 12, 1);
KbManagement.PublishingService.assignDraftTranslationTask(articleId, assigneeId,
instructions, dueDate, true);
```

cancelScheduledArchivingOfArticle(articleId)

Cancels the scheduled archiving of an online article.

Signature

```
public static Void cancelScheduledArchivingOfArticle(String articleId)
```

Parameters

articleId
Type: String

Return Value

Type: Void

Example

```
String articleId = 'Insert article ID';  
KbManagement.PublishingService.cancelScheduledArchivingOfArticle (articleId);
```

cancelScheduledPublicationOfArticle (articleId)

Cancels the scheduled publication of a draft article.

Signature

```
public static Void cancelScheduledPublicationOfArticle(String articleId)
```

Parameters

articleId
Type: String

Return Value

Type: Void

Example

```
String articleId = 'Insert article ID';  
KbManagement.PublishingService.cancelScheduledPublicationOfArticle (articleId);
```

completeTranslation (articleVersionId)

Puts a translation in a completed state that is ready to publish.

Signature

```
public static Void completeTranslation(String articleVersionId)
```


Parameters

articleVersionId
Type: String

Return Value

Type: Void

Example

```
String articleVersionId = 'Insert article ID';  
KbManagement.PublishingService.completeTranslation(articleVersionId);
```

deleteArchivedArticle(articleId)

Deletes an archived article.

Signature

```
public static Void deleteArchivedArticle(String articleId)
```

Parameters

articleId
Type: String

Return Value

Type: Void

Example

```
String articleId = 'Insert article ID';  
KbManagement.PublishingService.deleteArchivedArticle(articleId);
```

deleteArchivedArticleVersion(articleId, versionNumber)

Deletes a specific version of an archived article.

Signature

```
public static Void deleteArchivedArticleVersion(String articleId, Integer versionNumber)
```

Parameters

articleId
Type: String

versionNumber
Type: Integer

Return Value

Type: Void

Example

```
String articleId = 'Insert article ID';  
Integer versionNumber = 1;  
KbManagement.PublishingService.deleteArchivedArticleVersion(articleId, versionNumber);
```

deleteDraftArticle (articleId)

Deletes a draft article.

Signature

```
public static Void deleteDraftArticle(String articleId)
```

Parameters

articleId
Type: String

Return Value

Type: Void

Example

```
String articleId = 'Insert article ID';  
KbManagement.PublishingService.deleteDraftArticle(articleId);
```

deleteDraftTranslation (articleVersionId)

Deletes a draft translation.

Signature

```
public static Void deleteDraftTranslation(String articleVersionId)
```

Parameters

articleVersionId
Type: String

Return Value

Type: Void

Example

```
String articleVersionId = 'Insert article ID';  
KbManagement.PublishingService.deleteDraftTranslation (articleVersionId);
```

editArchivedArticle (articleId)

Creates a draft article from the archived master version and returns the new draft master version ID of the article.

Signature

```
public static String editArchivedArticle(String articleId)
```

Parameters

articleId
Type: String

Return Value

Type: String

Example

```
String articleId = 'Insert article ID';  
String id = KbManagement.PublishingService.editArchivedArticle(articleId);
```

editOnlineArticle (articleId, unpublish)

Creates a draft article from the online version and returns the new draft master version ID of the article. Also, unpublishes the online article, if *unpublish* is set to `true`.

Signature

```
public static String editOnlineArticle(String articleId, Boolean unpublish)
```

Parameters

articleId
Type: String

unpublish
Type: Boolean

Return Value

Type: String

Example

```
String articleId = 'Insert article ID';  
String id = KbManagement.PublishingService.editOnlineArticle (articleId, true);
```

editPublishedTranslation(articleId, language, unpublish)

Creates a draft version of the online translation for a specific language and returns the new draft master version ID of the article. Also, unpublishes the article, if set to `true`.

Signature

```
public static String editPublishedTranslation(String articleId, String language, Boolean  
unpublish)
```

Parameters

articleId

Type: String

language

Type: String

unpublish

Type: Boolean

Return Value

Type: String

Example

```
String articleId = 'Insert article ID';  
String language = 'fr';  
String id = KbManagement.PublishingService.editPublishedTranslation(articleId, language,  
true);
```

publishArticle(articleId, flagAsNew)

Publishes an article. If *flagAsNew* is set to `true`, the article is published as a major version.

Signature

```
public static Void publishArticle(String articleId, Boolean flagAsNew)
```

Parameters

articleId

Type: String

flagAsNew

Type: Boolean

Return Value

Type: Void

Example

```
String articleId = 'Insert article ID';
KbManagement.PublishingService.publishArticle(articleId, true);
```

restoreOldVersion(articleId, versionNumber)

Creates a draft article from an existing online article based on the specified archived version of the article and returns the article version ID.

Signature

```
public static String restoreOldVersion(String articleId, Integer versionNumber)
```

Parameters

articleId

Type: String

versionNumber

Type: Integer

Return Value

Type: String

Example

```
String articleId = 'Insert article ID';
String id = KbManagement.PublishingService.restoreOldVersion (articleId, 1);
```

scheduleForPublication(articleId, scheduledDate)

Schedules the article for publication as a major version. If the specified date is null, the article is published immediately.

Signature

```
public static Void scheduleForPublication(String articleId, Datetime scheduledDate)
```

Parameters

articleId

Type: String

scheduledDate

Type: Datetime

Return Value

Type: Void

Example

```
String articleId = 'Insert article ID';
Datetime scheduledDate = Datetime.newInstanceGmt(2012, 12, 1, 13, 30, 0);
KbManagement.PublishingService.scheduleForPublication(articleId, scheduledDate);
```

setTranslationToIncomplete(articleVersionId)

Sets a draft translation that is ready for publication back to “in progress” status.

Signature

```
public static Void setTranslationToIncomplete(String articleVersionId)
```

Parameters

articleVersionId

Type: String

Return Value

Type: Void

Example

```
String articleVersionId = 'Insert article ID';
KbManagement.PublishingService.setTranslationToIncomplete(articleVersionId);
```

submitForTranslation(articleId, language, assigneeId, dueDate)

Submits an article for translation to the specified language. Also assigns the specified user and due date to the submittal and returns new ID of the draft translation.

Signature

```
public static String submitForTranslation(String articleId, String language, String assigneeId, Datetime dueDate)
```

Parameters

articleId
Type: String

language
Type: String

assigneeId
Type: String

dueDate
Type: Datetime

Return Value

Type: String

Example

```
String articleId = 'Insert article ID';
String language = 'fr';
String assigneeId = 'Insert assignee ID';
Datetime dueDate = Datetime.newInstanceGmt(2012, 12,1);
String id = KbManagement.PublishingService.submitForTranslation(articleId, language,
assigneeId, dueDate);
```

INDEX

Q

query() call
SOQL overview [174](#)

S

SOQL
overview [174](#)
SOSL
about [182](#)