Search functions for KMC and Sphinx API

Last Modified on 01/26/2025 11:20 pm IST

> ⊛ **This article is designated for administrators.**

About

In the Kaltura platform, we utilize the Open Source Search Server Sphinx for metadata indexing and search functionality. To interact with Sphinx's search capabilities, we offer the Kaltura Search API and Custom Metadata Search API, which simplify the process.

This article is relevant for the KMC search behavior and commands or when using the Kaltura API with Sphinx.

Operators

In the Kaltura API's search field, you can use special symbols to make your searches more specific.

Search operator	What it means	Example
Exclamation mark (!)	**and not** - The entire phrase to the left of the ! will be partially searched. - The entire phrase to the right of the ! will be ignored even with an exact match. - A positive search word must appear before the **and** phrase. - The (!) operator, as well as double quotes for exact match, works on all filter text fields in the API.	Searching **Jane!Rockport** will return all entries that include a partial portion of the name, for example, **Jane**t, **Jan**o, **Ane**tte. The term **Rockport** will be ignored. NOTE: Searching for **!jane** alone will return an error.
Double quotes ("")	**an exact match**	Searching "**money ball"** will return all entries that contain the exact string **money ball** but not entries where **money** and **ball** were separated by other characters.
Backslash (\)	**an escape** (when used before certain characters, it changes the interpretation of those characters)	To search for a dot (**.**) in a string, you would need to 'escape' it like this: **\.**
Comma (,)	**or**	Searching **hello, world** finds entries with either **hello** or **world**.
Space	**and**	Searching for **hello world** will result in all entries that include both the word **hello** and the word **world**.

Double quotes in API string filtering

When generating APIs with filtering strings, the search will automatically add "" around each string before searching.This applies to strings with and without spaces in them, so adding quotation marks to strings will result in duplicated "" and might not retrieve the expected results.

Example: Filtering entries by tags using Kaltura API baseEntry.listAction and filtering by a "tagsNameMultiLikeOr" values:

✔ tagsNameMultiLikeOr = thisIstagwithoutspace , this is a tag with space

✘ tagsNameMultiLikeOr = **"thisIstagwithoutspace"** , **"this is a tag with space"**
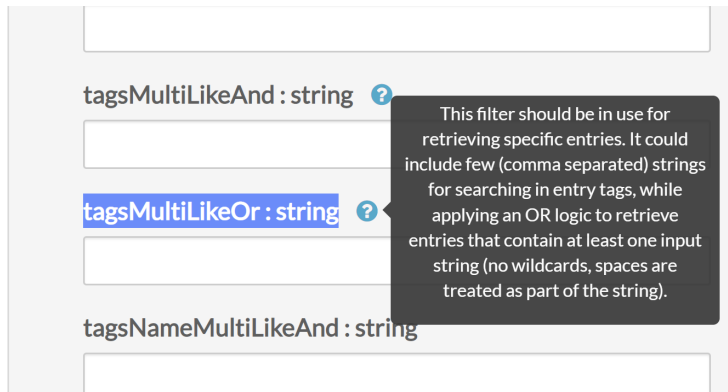
Blend characters

Blended characters are treated as both separators and valid characters in indexing. For example, let's say the **&** character is configured as blended, and the term **AT&T** appears in a document being indexed. In this case, three different keywords would be indexed: **at&t**, treating the blended character as valid, as well as **at** and **t**, treating **&** as a separator.

The following blend chars are configured for the API search.

!, $, ', (,), *, -, /, :, ;, <, =, #, [, ,], ^, `, {, |, }, ~, %, &, +, >, ?, @, _

These blend characters may be used as delimiters or as characters.

N-grams

When searching for up to **two** words, N-grams are used. N-grams are like a sliding window that moves across the word - a continuous sequence of characters of the specified length.

- Each word is broken into a 3 letter token.
- If there is a match of 80% - results are rendered.