Documentation

Documentation for the ConfiDoc Plugin for Confluence

ⓘ   See also tutorials for easy to follow examples on how to use ConfiDoc in different scenarios

TableView: *confidoc-table* ListView: *confidoc-list* CardView: *confidoc-card*	**The confidoc-table, confidoc-list, and confidoc-card** macros are used to present the REST service content (or RSS, CSV, XML) as a table, list (WYSIWYG), or card, each with filtering and data sorting/ordering.		
	Name of the service	servic eId	Services are defined by the Confluence administrator using the plugin's configuration screen
	Root element	root	Apply additional filters to the returned JSON data using field names as inputs
	Sorting /Ordering	sort	See examples below
	Filter	filter	Apply a filter on returned data; see examples below
	Cache resources locally	useCa che	Request results can be stored locally in the Confluence cache in order to improve page loading times and to reduce the number of requests to external servers; default local cache duration is 30 minutes
	Show service configuration dialog	enabl eLook up	Users will be able to configure their own parameters for a configured service request
	Define configuration parameters	lookup Labels	Define labels for the configuration dialog using key-value pairs, separated by "&" (as in URL notation), where <u>key</u> is the parameter name, and <u>value</u> is the text to be used as the label
	Caption for configuration button	lookup BtnLa bel	Provide a helpful pointer under the configuration dialog submit button
	Set HTML element ID	eleme ntId	Useful, when you want to have a predictable and constant element id in HTML for generated content. **I mportant: "_viewport" will be added to the name**

ConfiDoc Field	This macro defines a field to display returned data within the ConfiDoc TableView or ListView macros.		
confidoc-field	**Field name**	name	The name of the field in the returned data to be displayed by the TableView or ListView macros
	Field label	label	The label for the field to be displayed inside the table or list
	Field type	type	Choose the field type from these supported formats: • Link • Date • Number • Image • 'as-is' (blank: the value will be shown without transformation)
	Input format	inputFormat	Define the format in a returned data field in order to reformat using **Output format** (below), for example, to transform returned date values into more readable form
	Output format	outputFormat	Define the transformation rules for Output to be applied to data in the **Input format** (above)
	CSS	css	Define CSS rules to be applied on returned data in a rendered field
	Access control	restrictions	Control access to this field with a comma-separated list of group- or user names; leave blank if there are no restrictions
	Set HTML element ID	elementId	Useful, when you want to have a predictable and constant element id in HTML for generated content

ConfiDoc Excerpt *confidoc-excerpt*	Use this macro to include in the Confluence page the HTML or binary content (such as images) from an external webserver		
	Name of the service	serviceId	Services are defined by Confluence administrators with the ConfiDoc plugin configuration
	Selector	root	A CSS-like selector, but limited to a form: element attribute="value". To select elements by name, give the element's name; for example: **div** will select all the divs in the page To select all **div** elements in the page with the <u>myclass</u> class name, use: **div class="myclass"**
	Download and cache resources locally	download Resources	You can: • leave links to external resources "as is" • download external resources (such as images) to the Confluence server and adds them as attachments to the page • proxy external resources though local proxy
	Cache resources locally	useCache	Store requests locally in the Confluence cache to improve page-loading times and to reduce the number of requests to external servers (default cache duration is 30 minutes)
	Content Type	contentType	Force ConfiDoc to use the given content type
	Custom CSS styles	css	Custom CSS rules to apply on the rendered element
	Include as	type	This option provides a way to control how the macro outputs the returned data. Available types include: • Link • Date • Number • Image • 'as-is' (blank: the value will be shown without transformation)
	Show service configuration dialog	enableLookup	Users will be able to configure their own parameters for a configured service request
	Define configuration parameters	lookupLabels	Define labels for the configuration dialog using key-value pairs, separated by "&" (as in URL notation), where <u>key</u> is the parameter name, and <u>value</u> is the text to be used as the label
	Caption for configuration button	lookupBtnLabel	Provide a helpful pointer under the configuration dialog submit button
	Caption	download BtnLabel	When the content to be loaded is binary, a download link will be rendered; this parameter sets the label on that link
	Set HTML element ID	elementId	Useful, when you want to have a predictable and constant element id in HTML for generated content. Important: "_viewport" will be added to the name
ConfiDoc Excerpt Anchor *confidoc-excerpt-anchor*	This macro helps you to group content elements inside a block and easily reference this block later with the ConfiDoc Excerpt macro		
	Alphanumeric AnchorID (no spaces)	anchorId	Reference this block from the ConfiDoc Excerpt macro using the "div id=" AnchorID"' selector
ConfiDoc Show-For (IF) Macro *confidoc-show-for*	This macro displays included content only for given users and/or groups		
	Content is visible only to	restrictions	A comma-separated list of group names and/or user names

ConfiDoc Globals	This macro provides a convenient way to store and display global variables in Confluence		
confidoc-globals	**Variable name**	vars	Variable name

CSS Rules for ConfiDoc Fields	This is a helper macro to format fields visually based on their values		
confidoc-field-css	**Field value or expression**	con dition	Use values or expressions in order to match returned data much as you would for filters; ConfiDoc filters are very similar in functionality to ConfiForms Filters (also developed by Vertuna LLC)
	Field name	field Na me	Define the name of the field to which this rule should be applied (if blank, then the rule is applied to the row)
	CSS to be applied	css	CSS to apply when the above condition is met

Using filters

ConfiDoc filters employ syntax similar to the queries in Apache Lucene for expressions, which supports grouping using brackets (currently, ConfiDoc filters support only outer or single bracket sets, not inner or sub-brackets). Filters support matching for expressions that start with a wild-card, and filtering per field as well as for free text search (seeking a match against any field value).

The operators AND and OR are case-sensitive!

Filters support negative results with "not notation" to filter records that do not match the given filter with the '!'. For example: !f1:[empty] will search for records where the f1 field value is not empty (has some value).

Filters can perform simple arithmetic operations on dates (for example, to add or substract days from today's date). Here are some examples:

To match records where f1 contains 'success' and f2 is after yesterday or where f3 equals 'accepted':

(f1:*success AND f2:>[yesterday]) OR (f3:accepted)

To match records where f1 contains 'success' and where f2 is after yesterday or where f3 is not empty:

(f1:*success AND f2:>[today]-1) OR (!f3:[empty]) -

This filter would be invalid because ConfiDoc does not yet support inner brackets in expressions:

(f1:*success AND (f2:>[yesterday] OR (f3:accepted)).

Reserved words to use in expressions:

[empty]	To match when a particular field in the returned data is empty; for example: **field1:[empty]** will match records where field1 is empty (does not have a value even though the form defines this field)
[now]	To match on the current time and date; often useed with '<' and '>' for comparing with dates stored. For example: **someDateField:<[now]** will match records where the field 'someDateField' has a value which is in the past compared to now (current time)
[today]	To match on the current date, much like [now], but without hours and minutes
[tomorrow]	To compare the returned data against tomorrow's date (**[today]+1** could also be used)
[yesterday]	To compare the returned data against yesterday's date (**[today]-1** could be also used)
[end-of-month], [start-of-month], [now], [today], [yesterday], [tomorrow], [today]+N, [today]-N	Functions added / revised in ConfiDoc version 3.4.6
'<' and '>'	These can be used together with date and datetime fields, as well as to compare values in the returned data with values stored in numeric fields
'=<' and '>='	Added in ConfiDoc 3.4.6
!	To reverse or negate the filter condition; for example: **!field1:[empty]** will find records that have data in 'field1'

Additional examples:

- assuming field1 is of type date (or datetime), this filter will return records where the returned values of field1 are not older than 5 days before now:
  field1:[today]-5

- assuming field1 is of type date (or datetime), this filter will return records where the returned values of field1 is not after 10 days from now
  field1:[today]+10

Using sorting and limiting the number of records returned

ConfiDocs supports sorting results in ascending (ASC) and descending (DESC) directions. The syntax is similar to that in SQL, and the ordering rules must be coma separated.

Order directions ASC and DESC are case-sensitive!

Some examples: to limit the number of records returned, a LIMIT operator is used; for example,

f1 ASC, f2 DESC

will sort the dataset by the field f1 in <u>ascending</u> order and by f2 in <u>descending</u> order.

f1 ASC, f2 DESC LIMIT 2

will sort the dataset by the field f1 in <u>ascending</u> order and by f2 in <u>descending</u> order, and will limit the number of records returned to 2.

LIMIT 5

will limit the number of records returned to 5.