Toggle Post Filter, Menu – Shortcode Parameters

Toggle Post Filter (Pro) · https://publishpress.com/knowledge-base/toggle-post-filter-menu-shortcode-parameters/

The Toggle Post Filter is a frontend feature in the Tag Groups plugin. Visitors can select tags and see a list of posts that match their search criteria

This guide will show you how to use a shortcode to build the menu that controls the Toggle Post Filter. The menu displays the tags and allows users to make selections.

This shortcode can only be used once on a page (including widgets), and it cannot be used together with the shortcode or block of the Dynamic Post Filter.

Overview

Click here for an overview of all shortcodes for the Toggle Post Filter.

Data Types

- **string:** A text. For identifiers that won't be visible to visitors you often have to replace spaces with _ or -. If the string contains spaces, you need to wrap it in quotes.
- **integer:** A number without decimal places.
- **0 or 1**: 0 turns the feature off and 1 on.

Parameters

Tags and Terms

hide_empty

Hide tags that are not being used in (public) posts.

accepted values	default	example shortcode
integer	1 (on)	`hide_empty=0`

operator

The operator determines how selected tags and groups should be logically connected when searching for matching posts. Possible values are "IN"/"OR" (at least one tag is among the post tags), "AND" (all tags are among the post tags of the same group), "IN AND" (at least one tag from each group is among the post tags), "EXACT" (the selection of tags must be identical to the post tags of that same group, not more and not less), or a per-group definition:

```
operator="all:OPERATOR|group ID:OPERATOR|group ID:OPERATOR| ..."
```

Remarks about the per-group notation:

- You need at least one entry with the keyword "all". The corresponding operator determines how the groups are connected.
- *OPERATOR* can be AND, OR or EXACT (upper or lower case). IN is an alias of OR.
- The order of groups doesn't matter.
- If you use this format of the operator, all groups that you don't specify will fall back to the operator OR.

Note for the operator "exact": If you upgrade from a version before 1.39.0, go to the settings, Troubleshooting and run the maintenance.

accepted values	default	example shortcode

string	IN/OR	`operator="IN AND"` `operator="all:and|1:or|5:exact|3:and"`

preset_tags

See the information under Preset Tags.

accepted values	default	example shortcode
string	none	`preset_tags="tag-slug-1,tag-slug-2"`

term_orderby

Lets you determine the order of the tags (terms). You can modify the tags' slugs or descriptions in order to achieve the required order.

option	description
name	tag name
natural	natural sorting, e.g. "3" before "12"
slug	see on the Tags page
term_id	term ID
description	see on the Tags page
count	post count
term_order	A custom sort order is available with a 3rd-party plugin.

accepted values	default	example shortcode
string	name	`term_orderby=natural`

term_order

Whether to sort the tags in ascending or descending order.

accepted values	default	example shortcode

string ("asc" or "desc", capitalization doesn't matter)	ASC	`term_order=asc`

taxonomy

Restrict the tags only to these taxonomies. Available are only taxonomies that have been activated in the Settings.

accepted values	default	example shortcode
comma-separated list of taxonomy names	Taxonomies selected in the settings under "Basics".	`taxonomy=post_tag,product_tag`

include_terms

Select terms (tags) that should be included. Empty means all.

accepted values	default	example shortcode
comma-separated list of integers	empty (=all)	`include_terms=1,5,9`

exclude_terms

Select terms (tags) that should be excluded. Empty means none.

accepted values	default	example shortcode
comma-separated list of integers	empty (=none)	`exclude_terms=1,5,9`

static_taxonomy

Optionally filter also by terms (tags, categories, ...) of an additional taxonomy (not necessarily one of the taxonomies used for tag groups). That way you can narrow down the total sample of filtered posts.

Used together with static_terms.

accepted values	default	example shortcode
string	none	`static_taxonomy=category`

static_terms

Used together with static_taxonomy. This parameter lists the IDs of the additional tags, categories, ... that should be present in all posts used by the filter.

accepted values	default	example shortcode
comma-separated list of integers	none	`static_terms=42,255`

Groups

accordion

Display the menu as an accordion. The accordion opens when clicking on a group label. use accordion=2 to open on mouseover (i.e. when hovering your mouse over a label).

Please note that also hidden toggle switches are effective.

accepted values	default	example shortcode
integer	0 (off)	`accordion=1`

include

Tag groups that will be shown to visitors. Groups without tags will be hidden.

In the short code you use the IDs, which you find in the left column in the list of groups. Empty or not used means that all tag groups will be used.

accepted values	default	example shortcode
comma-separated list of integers	empty (=all)	`include=1,5,9`

one_only_groups

Tag groups where visitors can select only one tag. All of these groups must of course be available in the filter.

In the short code you use the IDs, which you find in the left column in the list of groups. Please note that more tags might still be selected if this selection is carried over by the persistent filter setting from another page.

accepted values	default	example shortcode
comma-separated list of integers	empty (=none)	`one_only_groups=1,5,9`

Layout and Theme

layout

- **classic:** The menu items appear vertically stacked in a narrow container. Posts have space to display on the right side. If you select "classic" for the menu, also the body part with the posts should use the "classic" layout.
- **button:** Like the classic layout, but instead of toggles we use buttons. Optionally show an icon on active buttons.
- **classic_tags:** The menu is arranged like the classic layout. Tags appear in a tag shape and line up in rows (like in tag clouds).
- **wide:** Groups are lined up horizontally. Posts should display below the menu. If you select "wide", the post layout can be any of wide, boxed, columns, columns-avoid-break and masonry.
- **wide_button:** Like the wide layout, but instead of toggles we use buttons. Optionally show an icon on active buttons.
- **wide_tags:** The menu is arranged like the wide layout. Tags appear in a tag shape and line up in rows (like in tag clouds).
- **slider_left:** The menu slides in from the left side, covering the contents of the page. The slider remains open while you click on the buttons so that you scroll through the search results while the menu remains open. Consider composing the page in a way that the slider doesn't cover the posts. You close the menu by clicking on the X or outside the slider.
  For the slider you will also need the button to open the slider.
  The default slider uses buttons with optional icons.
- **slider_left_tags:** The menu is arranged like the slider_left layout. Tags appear in a tag shape and line up in rows (like in tag clouds).
- **slider_right:** The same as the left slider, opening from the right.
- **slider_right_tags:** The menu is arranged like the slider_right layout. Tags appear in a tag shape and line up in rows (like in tag clouds).

accepted values	default	example shortcode
string	classic	`layout=wide`

icon_class

Use the identifier (the part in lowercase letters with dashes) to place an icon on active buttons. (Not available with toggles.)

often used: dashicons-tag, dashicons-yes, dashicons-yes-alt, dashicons-heart, dashicons-arrow-left, dashicons-arrow-right, dashicons-thumbs-up

accepted values	default	example shortcode
string	empty	`icon_class=dashicons-tag`

slider_width

Set the width of the slider menu in pixels. Only available with the slider_left_tags and the slider_right_tags menu layouts.((In the other slider layouts the width is fixed.))

accepted values	default	example shortcode
integer	600	`slider_width=300`

tag_color

Set the background color of the tags. Only available with the classic_tags, wide_tags, slider_left_tags and the slider_right_tags menu layouts.

accepted values	default	example shortcode
string	600	`tag_color="#ddd"`

selected_tag_color

Set the background color of the tags that are toggled on in the filter. Only available with the classic_tags, wide_tags, slider_left_tags and the slider_right_tags menu layouts.

accepted values	default	example shortcode

string	600	`selected_tag_color="#d00"`

theme

Available are "light" and "dark".

accepted values	default	example shortcode
string	light	`theme="dark"`

Caching and Performance

caching_time

Time in minutes that results remain in the cache.

accepted values	default	example shortcode
integer	10	`caching_time=10`

persistent_filter

The Toggle Post Filter can optionally remember the selection of groups and tags so that your visitors who clicked on an article and then return to this page will see the same list of articles again. If now post has been clicked, the page will scroll to the group selection menu. (Otherwise the browser might go to a random post, depending on the loading speed.)

The data is saved in a browser cookie. If you are using this filter more than once on your site, you should use different values for div_id so that each filter uses a different cookie.

The value is the time in minutes. Zero means off.

accepted values	default	example shortcode
integer	0 (off)	`persistent_filter=30`

timeout

We try to avoid search requests each time a visitor toggles a button or types a letter in the text field because the search should be launched only when the visitor is done selecting tags and typing text.

We achieve this through a timeout of inactivity before the request is sent to the server. This timeout is by default 1000ms (= 1 second) and we dynamically increase it (plus up to 2 seconds) when we assume that a visitor needs more time selecting menu options. We reset this increased timeout when the posts load.

You can set here the (minimum) timeout to find a good balance between user experience (with a responsive search) and database load. Factors to be considered are the habits and needs of typical visitors, the number of visitors who simultaneously search, the size of your database (posts and tags) and the power of your database engine.

accepted values	default	example shortcode
integer (milliseconds)	1000	`timeout=500`

Text Search

text_search

Enable text search and determine when the search should start.

0: off
1: launch by pressing enter
2: launch by pressing enter or pause typing

accepted values	default	example shortcode
integer	0	`text_search=2`

Labels and Messages

placeholder_text_search

Placeholder message for the text search field.

accepted values	default	example shortcode
string	"type here"	`placeholder_text_search="try word or -word"`

Advanced Styling

You can set here classes to be referred to in CSS. You need some knowledge of CSS to use these options.

div_class

Define a class for the .

accepted values	default	example shortcode
string	dpf_toggle_menu_light	`div_class="dpf_toggle_menu_dark"`

div_id

Define an ID for the enclosing .

accepted values	default	example shortcode
string	empty	`div_id="my_own"`