

Add or disable option in select list, radio button, multi-select list or checkbox

This function has been **renamed** with the [JWT 3.0](#) release.

Find the new documentation at:

[Add or disable option in \(multi-\) select list, radio button, or checkbox field](#)

On this page

- [Purpose](#)
- [Configuration Parameters](#)
- [Usage Examples](#)
- [Related Features](#)

Purpose

Add or Disable option in Select List, Radio Button, Multi-Select List or Checkbox post-function allows to dynamically **adding new options, re-enabling disabled options** and **disabling existing options** in custom fields with single or multiple selectable options. Supported custom field types are:

- Select List
- Radio Button
- Multi-Select List
- Checkboxes

Configuration Parameters

This screenshot shows configuration parameters for the post-function:

Target field: Select List, Radio Button, Multi-Select List or Checkbox custom field that will receive the new option.	Classification - [Select List (single choice)] ▼
Option Name: Source field containing the name of the new option to be inserted or disabled.	Ephemeral string 1 - [Text] ▼
Action: Action to be carried out by the post-function on the option with input name.	☒ Insert new option, or do nothing if it already exists. ☐ Insert new option, or re-enable it if it already exists and is disabled . ☐ Insert new option, or insert a duplicated one if it already exists. ☐ Insert new option, or insert a duplicated one if it already exists and is disabled . ☐ Disable existing option.
Sort options alphabetically:	☐ If checked, options are sorted alphabetically, otherwise the new option is added at the end of the options list.
Conditional execution: Optional boolean expression that should be satisfied in order to actually execute the post-function. (Syntax Specification)	1 Leave the field empty for executing the post-function unconditionally. Collection of Examples [Line 1 / Col 1] <u>Logical connectives:</u> and, or and not. Alternatively you can also use &, and !. <u>Comparison operators:</u> =, !=, >, >=, < and <=. Operators in, not in, any in, none in, ~ and != can be used with <i>strings</i>, <i>multi-valued fields</i> and <i>lists</i>. <u>Logical literals:</u> true and false. Literal null is used with = and != to check whether a field is initialized, e.g. {00012} != null checks whether <i>Due Date</i> is initialized. String Field Code Injector: Summary - [Text] - %{00000} ▼ Numeric/Date Field Code Injector: Original estimate (minutes) - [Number] - {00068} ▼ Check Syntax

Ephemeral string virtual fields are typically used to store the name of the new option to be added or re-enabled. These fields are previously set by post-function [Copy parsed text to a field](#).

Note: **Disable option** feature is available since version **2.2.26**.

Usage Examples

Related Features

- [Add or disable option in cascading select](#)