11.4 Process SDK Registration Procedure

The procedure described below is the same for the 3 types of process sdk functions.

Suppose we need to create and register a “FIELD ACTION” type SDK function that allows you to perform the division between multiple values.
Once written, **it must be inserted into the Utils.php file available in the modules/SDK/src/ProcessMaker path**.
It must then be registered through the following command to be inserted into the script.php file available in the plugins/script path:

SDK::setProcessMakerFieldAction('function name','path','function label');

'function name' → name of the function
'path' → path where the function to be registered is located
'function label' → name with which the function will be displayed to the user in the processes section.

We then insert the command in the script.php file and compile it with the data from the SDK:

SDK::setProcessMakerFieldAction('division','modules/SDK/src/ProcessMaker/Utils.php','division(value1,value2,...)');

N.B: Each typology has its own dedicated command, so if we had to load an “ACTION” type SDK we should use the following command:
SDK::setProcessMakerAction('function name','path','function label');

If instead we had to load an “TASK CONDITION” type SDK we should use the following command:
SDK::setProcessMakerTaskCondition('function name','path','function label');

Once these steps are completed, just launch the script.php file and the function will be correctly registered.

Revision #2
Created 2025-05-27 10:56:18 UTC by Riccardo
Updated 2026-04-13 10:43:31 UTC by Tommaso