



# ALTAIR

Altair<sup>®</sup> FluxMotor<sup>®</sup> 2022.2

Script Factory

General user information

## Contents

<b>1</b>	<b><i>Script Factory overview</i></b>	<b>3</b>
1.1	Main areas of Script Factory	3
1.2	How to get into Script Factory?	3
1.3	Advice for use	4
<b>2</b>	<b><i>Python file management</i></b>	<b>5</b>
2.1	New file	5
2.2	Edit a file	6
2.3	Duplicate a file	7
2.4	Delete a file	8
2.5	Store a file	9
2.6	Save a file	10
2.7	Run and Stop	11
2.8	Editor	12
2.8.1	Editor main functionality	12
2.8.2	FluxMotor command help files	12
2.8.3	Log and error	13
2.8.4	Open a file in system explorer	14
<b>3</b>	<b><i>Miscellaneous topics</i></b>	<b>15</b>
3.1	Script for filling the slots	15
3.2	Feasibility of the winding architecture and the slot filling	16

# 1 SCRIPT FACTORY OVERVIEW

## 1.1 Main areas of Script Factory

“Script Factory” is a dedicated application to create and manage python Altair® FluxMotor® scripts.

The application includes:

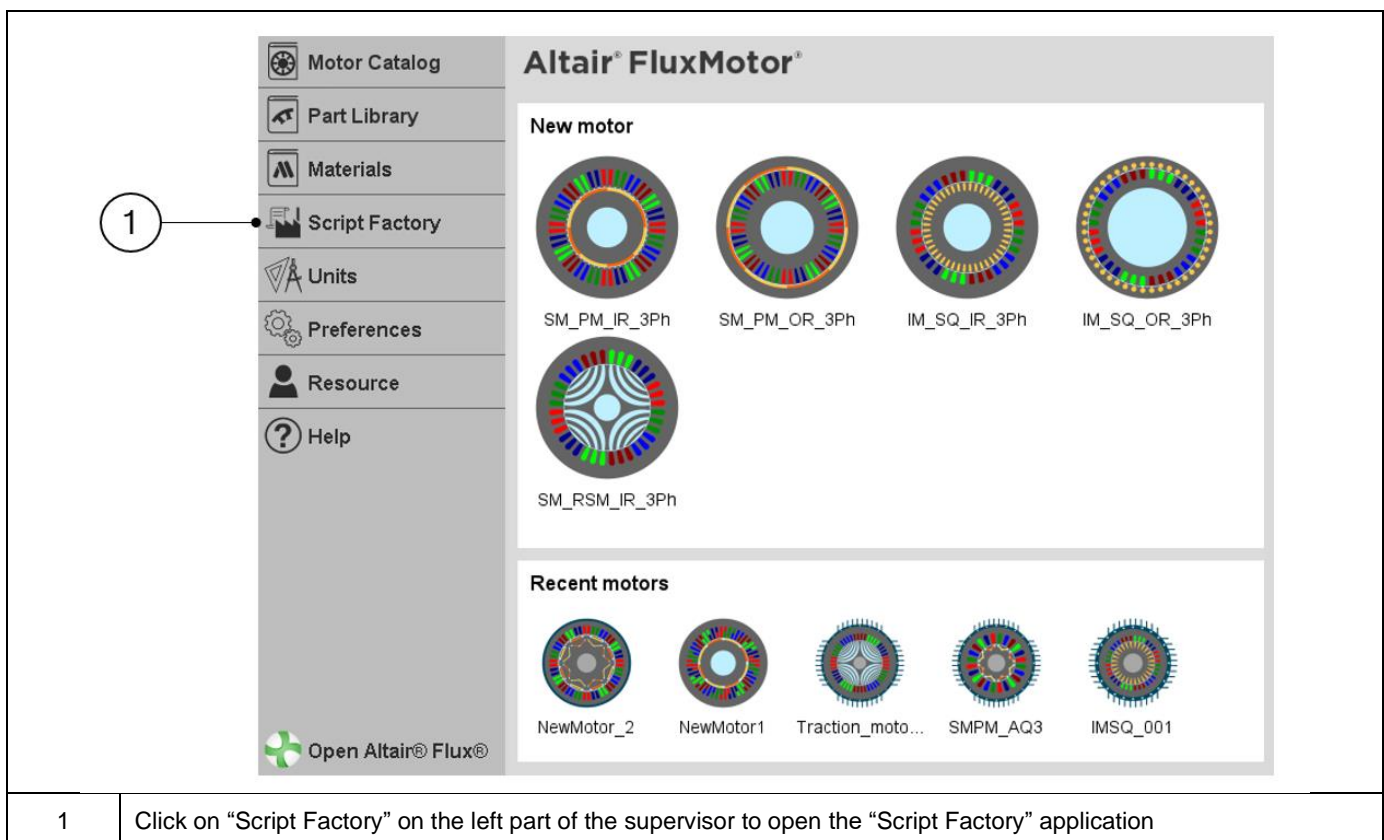
- Workspace in tree format
- Script files functionality as “New, Duplicate and Delete”
- Launcher of python script with “Run and Stop” function

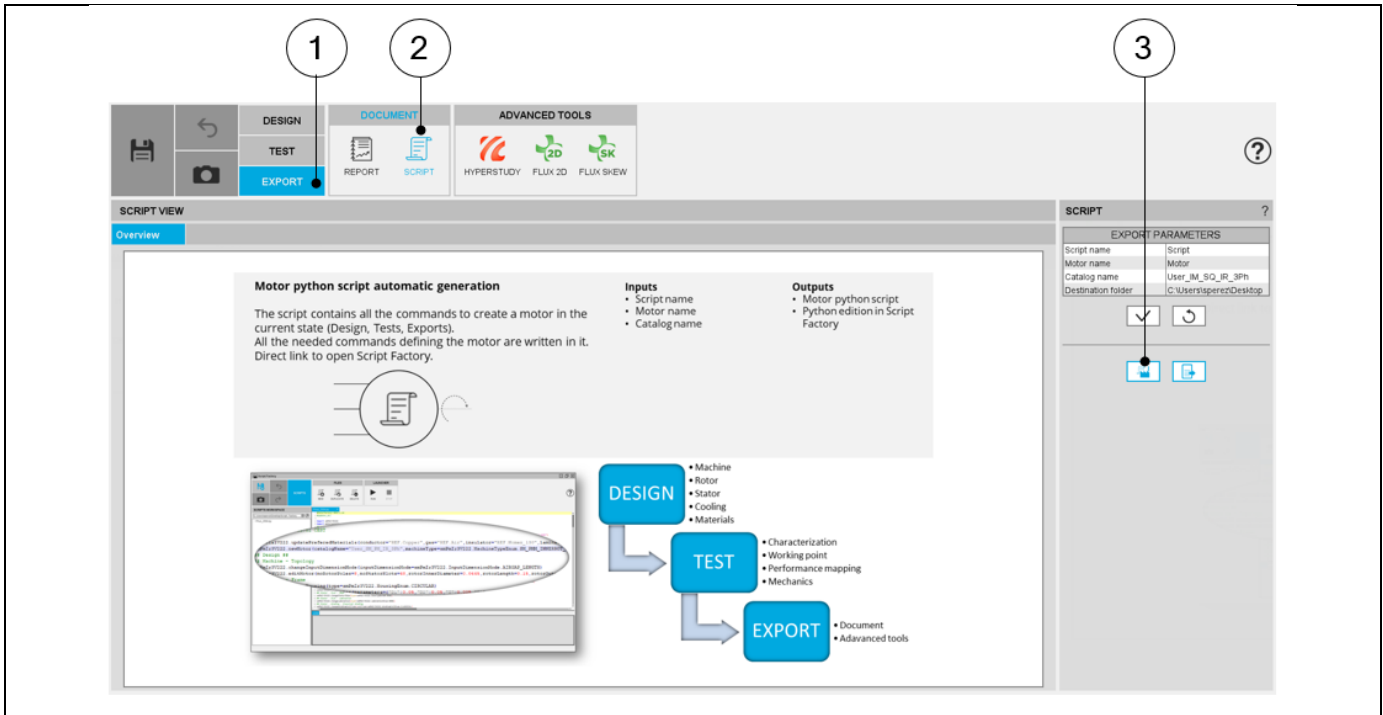
Classical functions such as “Save, Undo, Redo and Store” are also available.

## 1.2 How to get into Script Factory?

Two ways are possible:

- 1) From Supervisor, click on “Script Factory” button
- 2) From Motor Factory / Export environment / Script, it is possible to open the python script of a current motor with all the needed commands defining it.





### How to get into “Script Factory” from Motor Factory?

1	In Motor Factory, select EXPORT environment
2	In EXPORT environment, select SCRIPT in the DOCUMENT menu
3	In “SCRIPT” panel of Motor Factory, open the python script of the current motor by clicking on this button

## 1.3 Advice for use

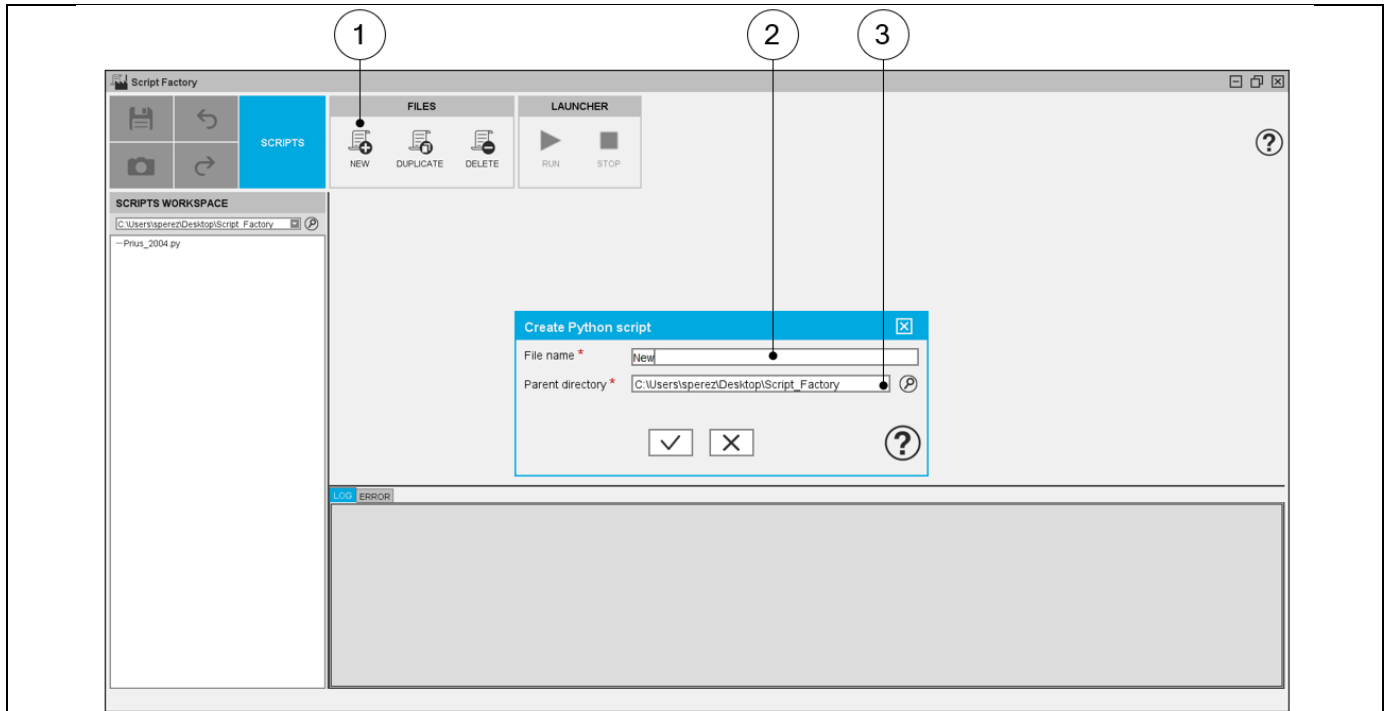
Altair® FluxMotor® is a dedicated platform, which can be used for the predesign of electrical motors. The target of Altair® FluxMotor® is to get a quick overview of technical and economic potential of motors.

The motive of the Script Factory is to give the possibility to automate some study such as lunch serial tests or serial design winding configurations.

## 2 PYTHON FILE MANAGEMENT

### 2.1 New file

In Script Factory application, a new python file can be created and stored in the selected workspace.

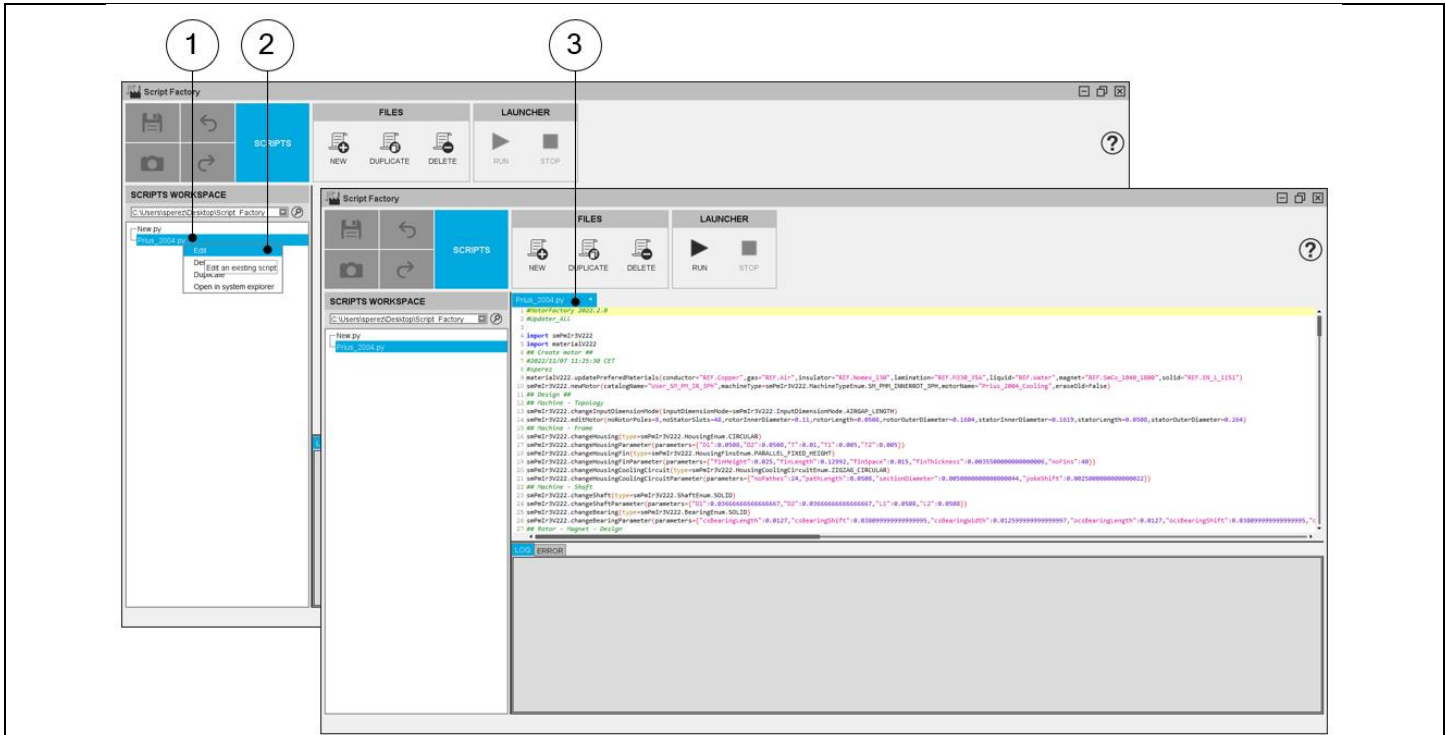


#### How to create a new python Script?

1	By clicking on the icon - NEW in the FILES menu.
2	Give a name to the python script.
3	Select a parent working directory (The default one is the current workspace).

## 2.2 Edit a file

In Script Factory application, a selected python file from the workspace tree can be edited.



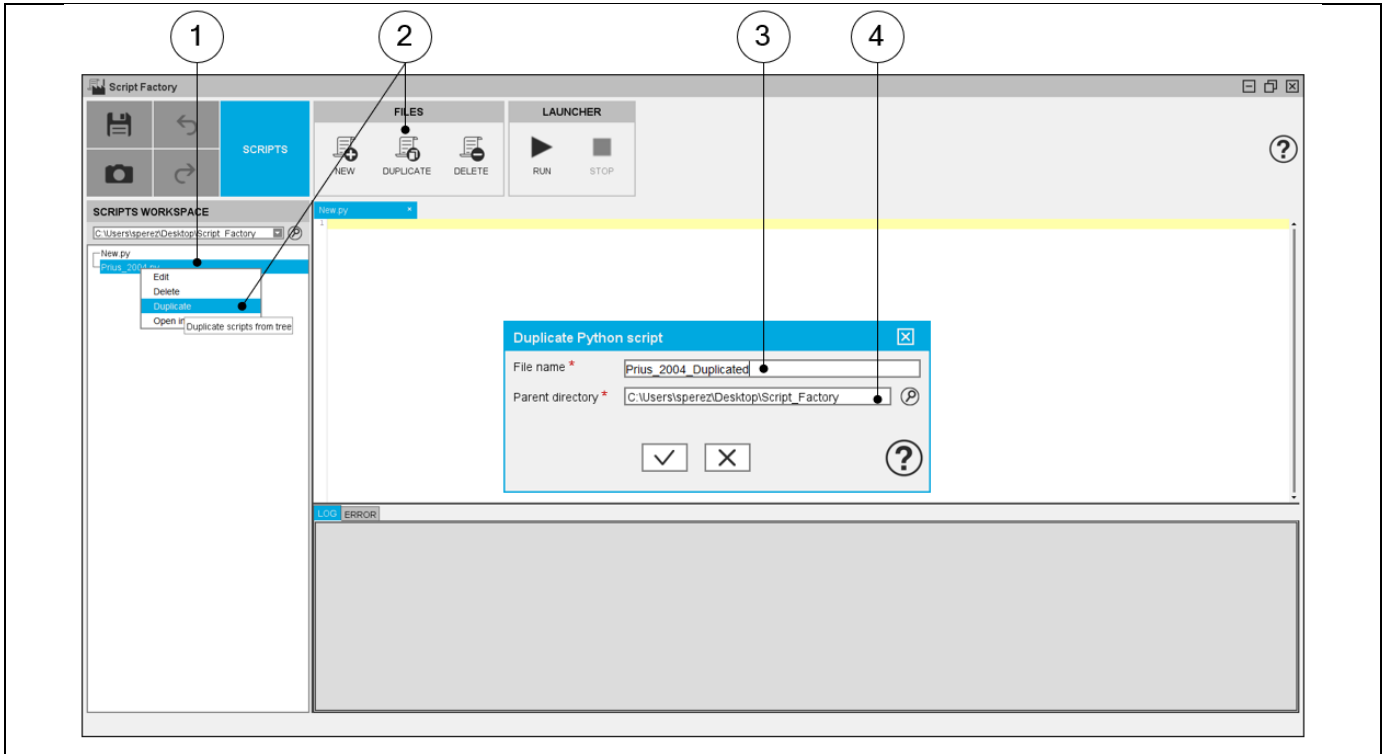
How to edit a python Script from the workspace tree?

1	Select a python file in the workspace tree.
2	Right click on the selected python file and select "Edit" option.
3	The python script is edited in the editor.

Note: A double click on the python file directly open it in the editor.

## 2.3 Duplicate a file

In Script Factory application, a selected python file from the workspace tree can be duplicated in a parent directory.

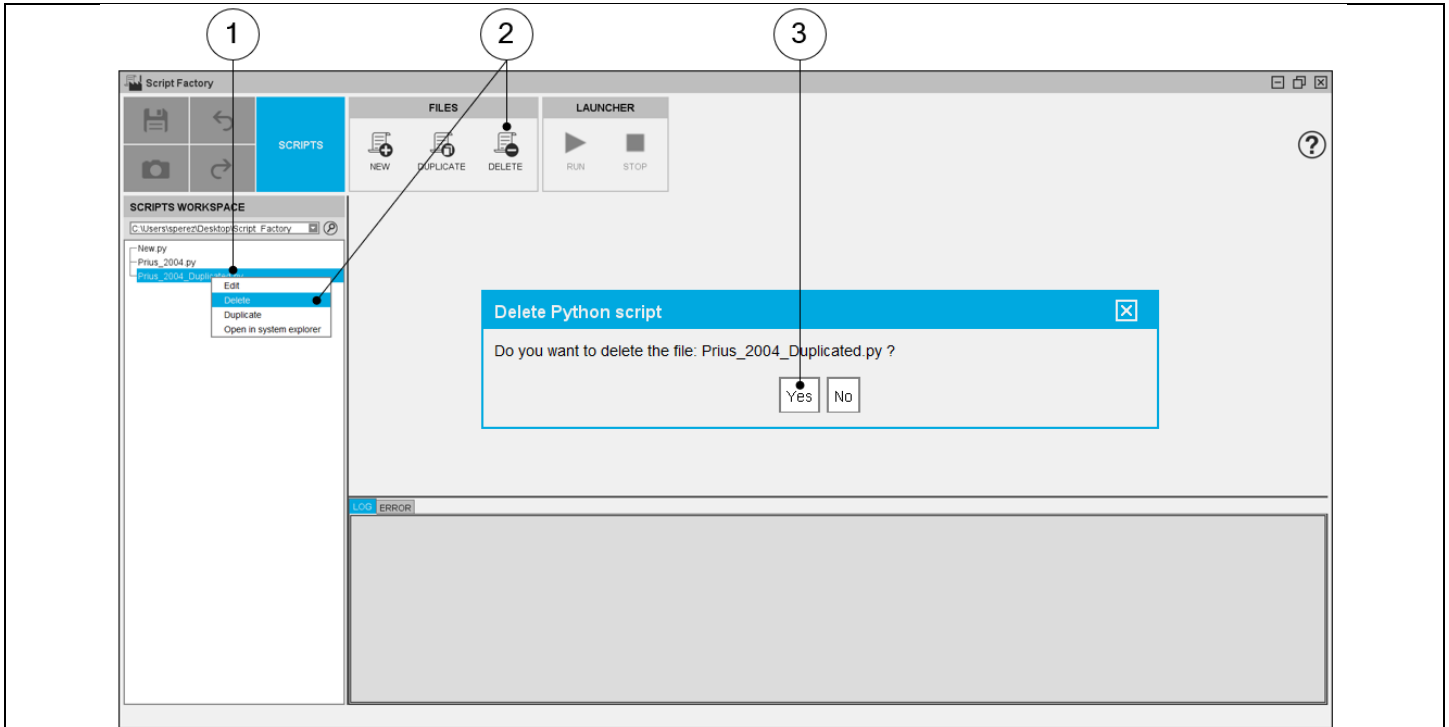


How to duplicate a python Script from the workspace tree?

1	Select a python file in the workspace tree.
2	Click on the icon DUPLICATE in the FILES menu or right click on the selected python file and then select the "Duplicate" option.
3	Give a name to the python script.
4	Select a parent working directory (The default one is the current workspace).

## 2.4 Delete a file

In Script Factory application, a selected python file from the workspace tree can be deleted.



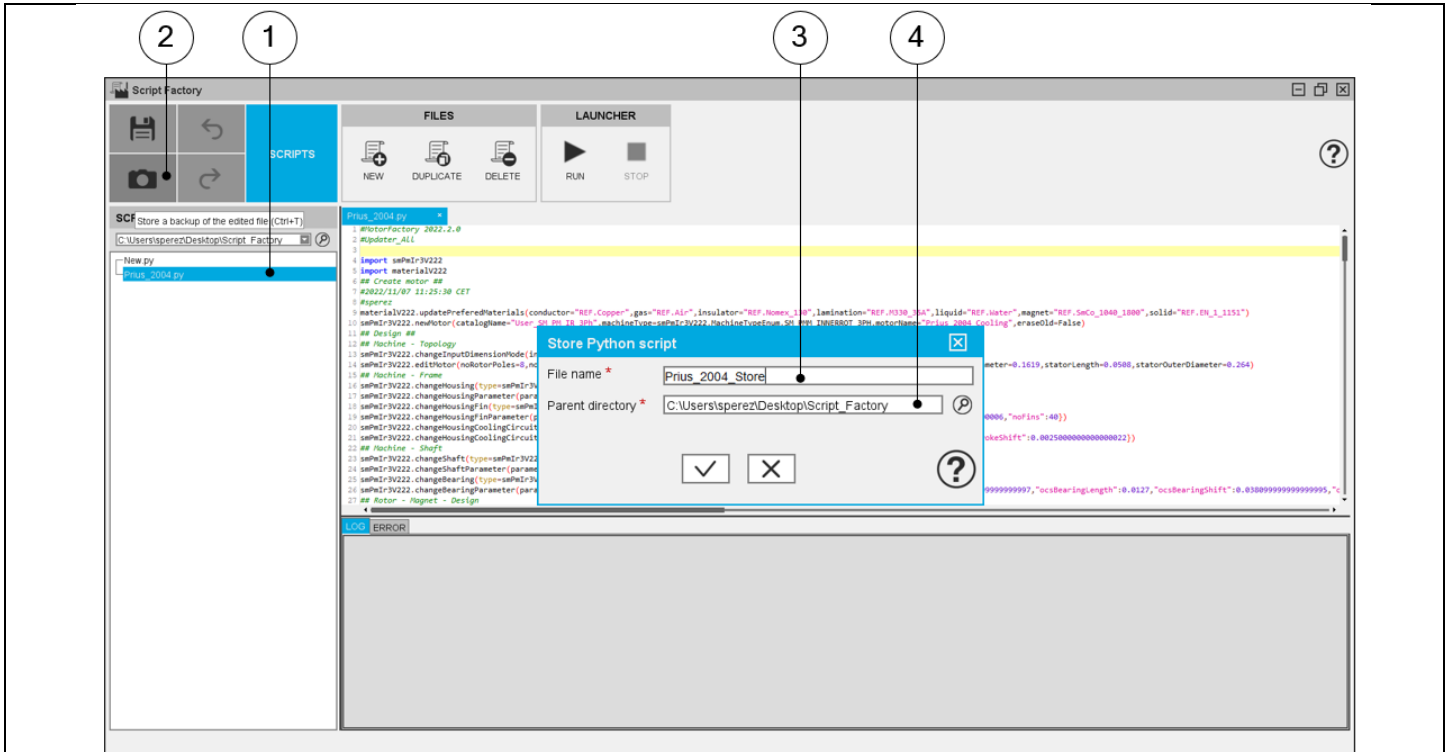
How to delete a python Script from the workspace tree?

1	Select a python file in the workspace tree.
2	Click on the icon DELETE in the FILES menu or right click on the selected python file and then select the "Delete" option.
3	Click on "Yes" to confirm the file delete.



## 2.5 Store a file

In Script Factory application, a selected python file from the workspace tree can be stored.



### How to store a python Script from the workspace tree?

1	Select a python file in the workspace tree.
2	Click on the icon STORE via the top expanded menu.
3	Give a name to the python script you want to store.
4	Select a parent working directory (The default one is the current workspace).

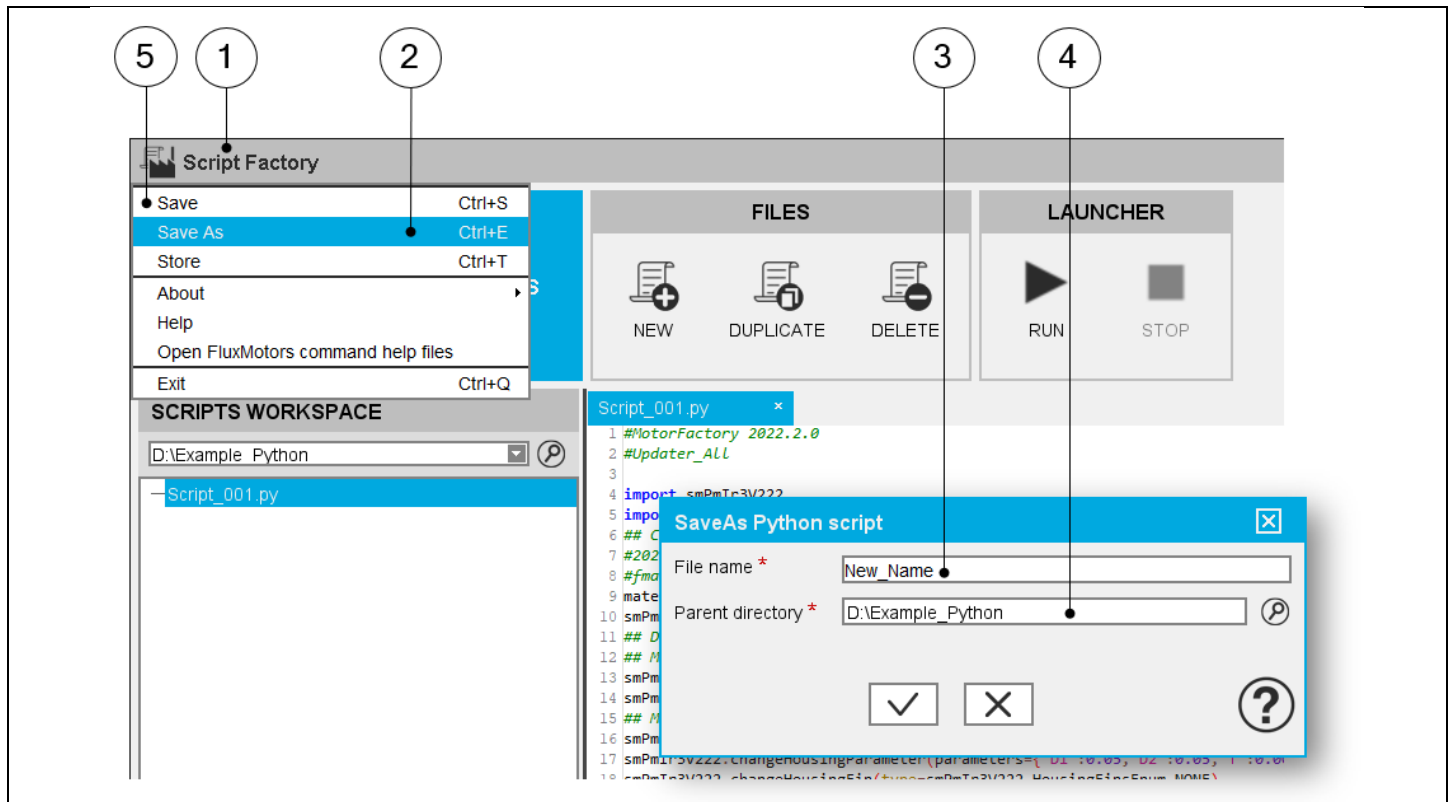
#### Note:

- To “Store” a file you can also use the classical shortcut keyboard “Ctrl+T”

## 2.6 Save a file

Save As and Save functions:

In Script Factory application, an edited python file can be saved with a new name (Save As function) or with the same name (Save function).



How to “Save As” a python Script ?

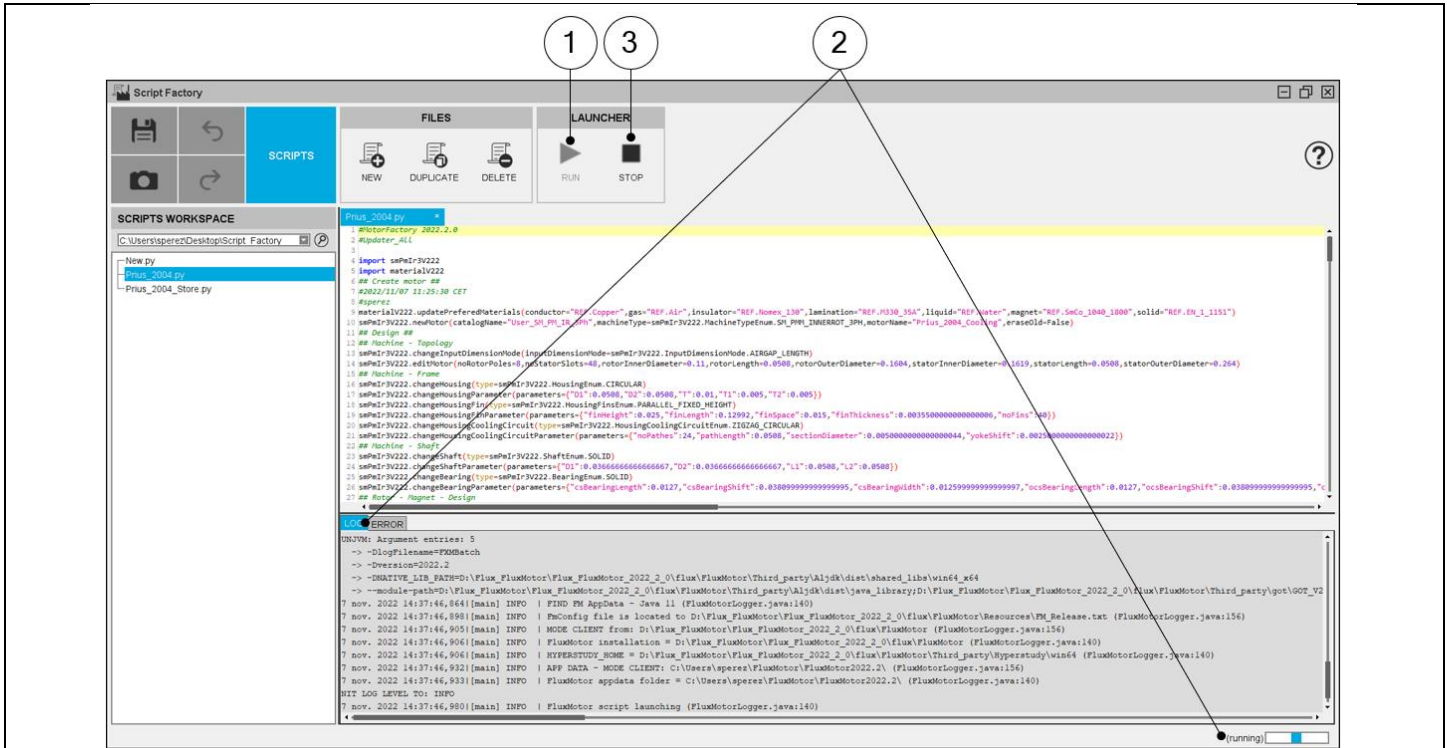
1	Click on the dropdown menu select the function “Save As”
2	Click on the function “Save As” via the dropdown menu.
3	Give a name to the python script you want to save.
4	Select a working folder to store the new file (The default one is the current workspace).
5	A python file can be saved with the same name with the “Save” function.

Note:

- To “Save” a file you can also use the classical shortcut keyboard “Ctrl+S”
- To “Save as” a file you can also use the classical shortcut keyboard “Ctrl+E”

## 2.7 Run and Stop

In Script Factory application, there is a launcher of python file, that means an edited python file can be run and stop.



### How to run and stop a python Script from the workspace tree?

1	Click on the icon RUN via the LAUNCHER menu to run the Edited file.
2	The LOG folder scrolls during the run and a ping-pong bar appears.
3	To stop the file execution, click on the icon STOP via the LAUNCHER menu.

## 2.8 Editor

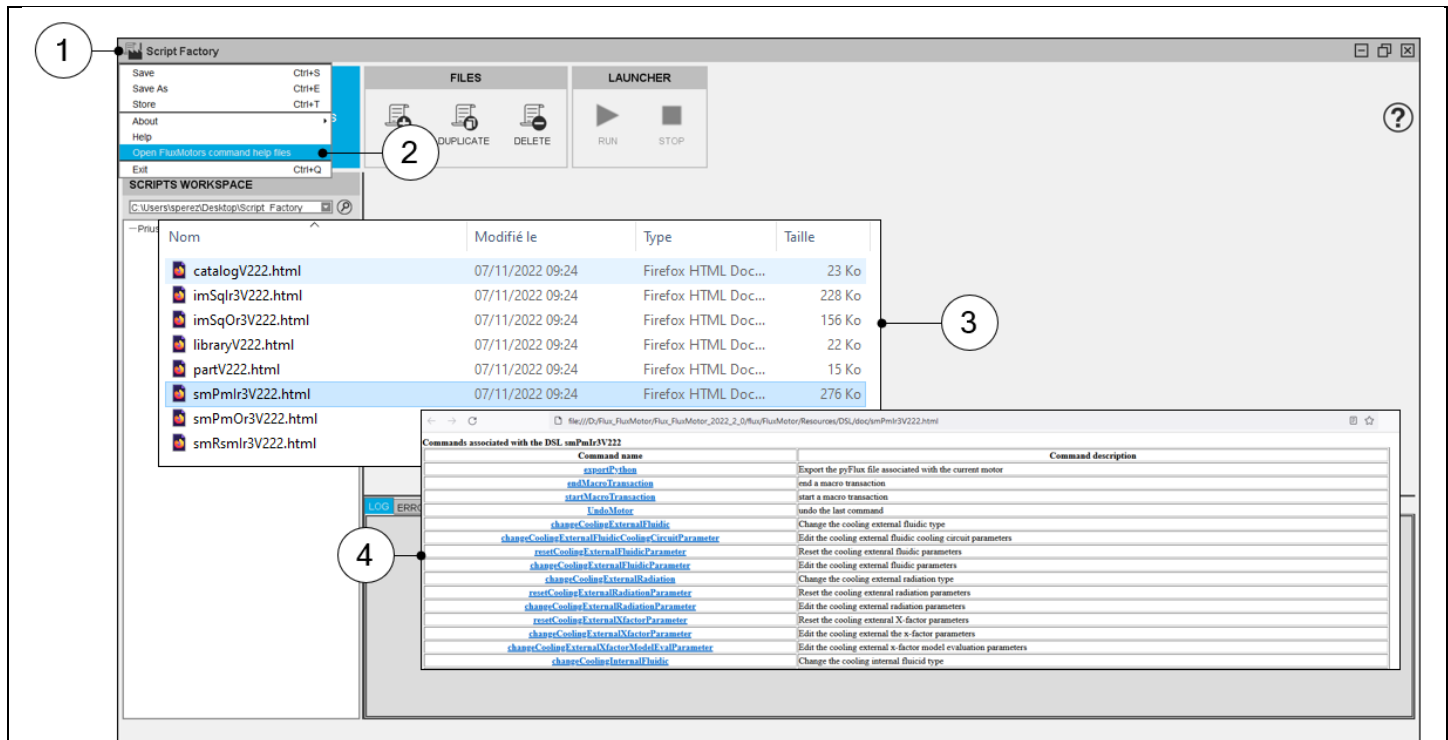
Script Factory editor is located at the center panel of the application, and it allows to edit the python script in the usual way.

### 2.8.1 Editor main functionality

The editor proposed the following basic functionality through shortcut keyboard:

- “Save” a file - “Ctrl+S”
- “Save as” a file - “Ctrl+E”
- “Store” a file - “Ctrl+T”
- “Find and replace” a string - “Ctrl+F”
- “Undo” an action - “Ctrl+Z”
- “Redo” an action - “Ctrl+Y”
- “Close” the application Script Factory - “Ctrl+Q”

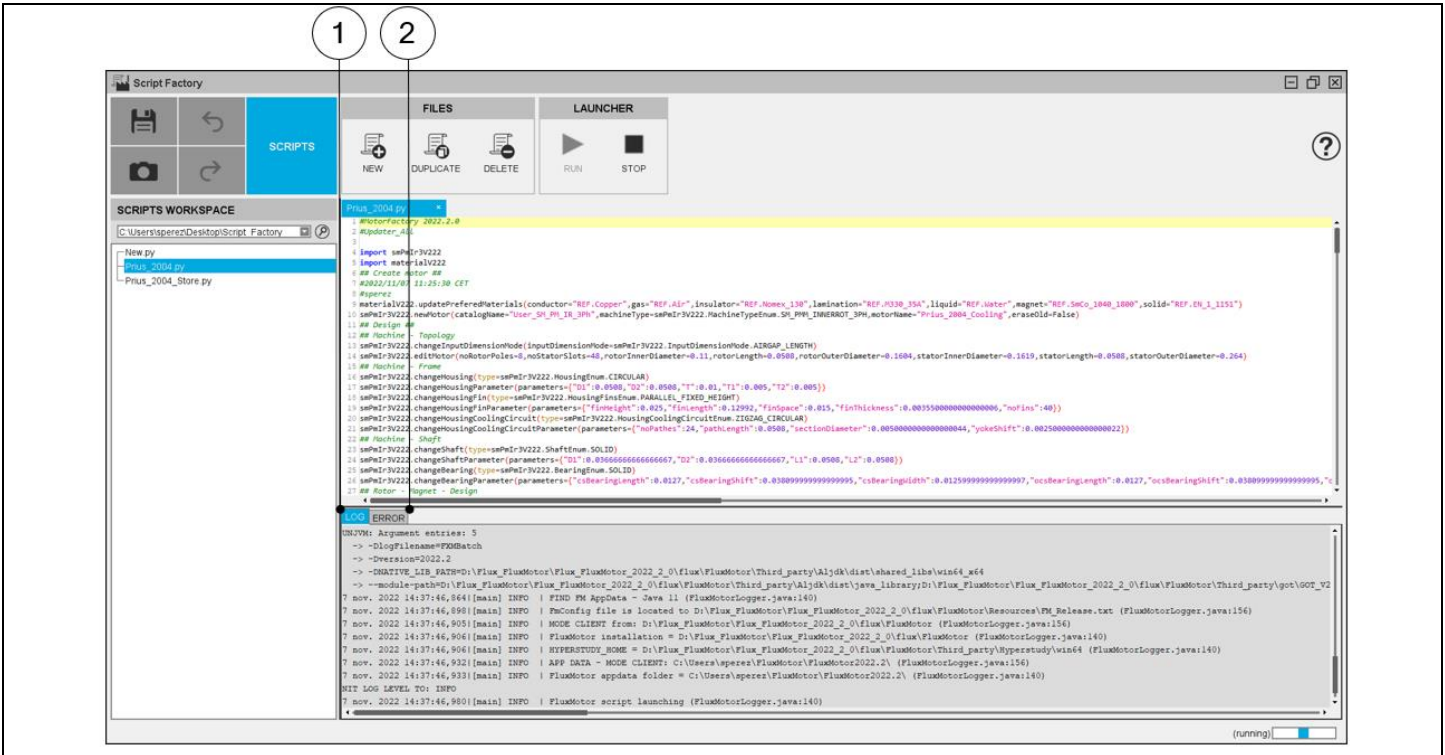
### 2.8.2 FluxMotor command help files



#### How to access to FluxMotor python script command help?

1	Click on the icon “Script Factory” on the top left dropdown menu.
2	Select “Open FluxMotor command help files” option.
3	List of available DSL (Domain Specific Language) commands dedicated to the main applications of FluxMotor. One must select the one in which are stored the targeted commands.
4	The command list is displayed. One must click on the command name to see the corresponding description.

### 2.8.3 Log and error

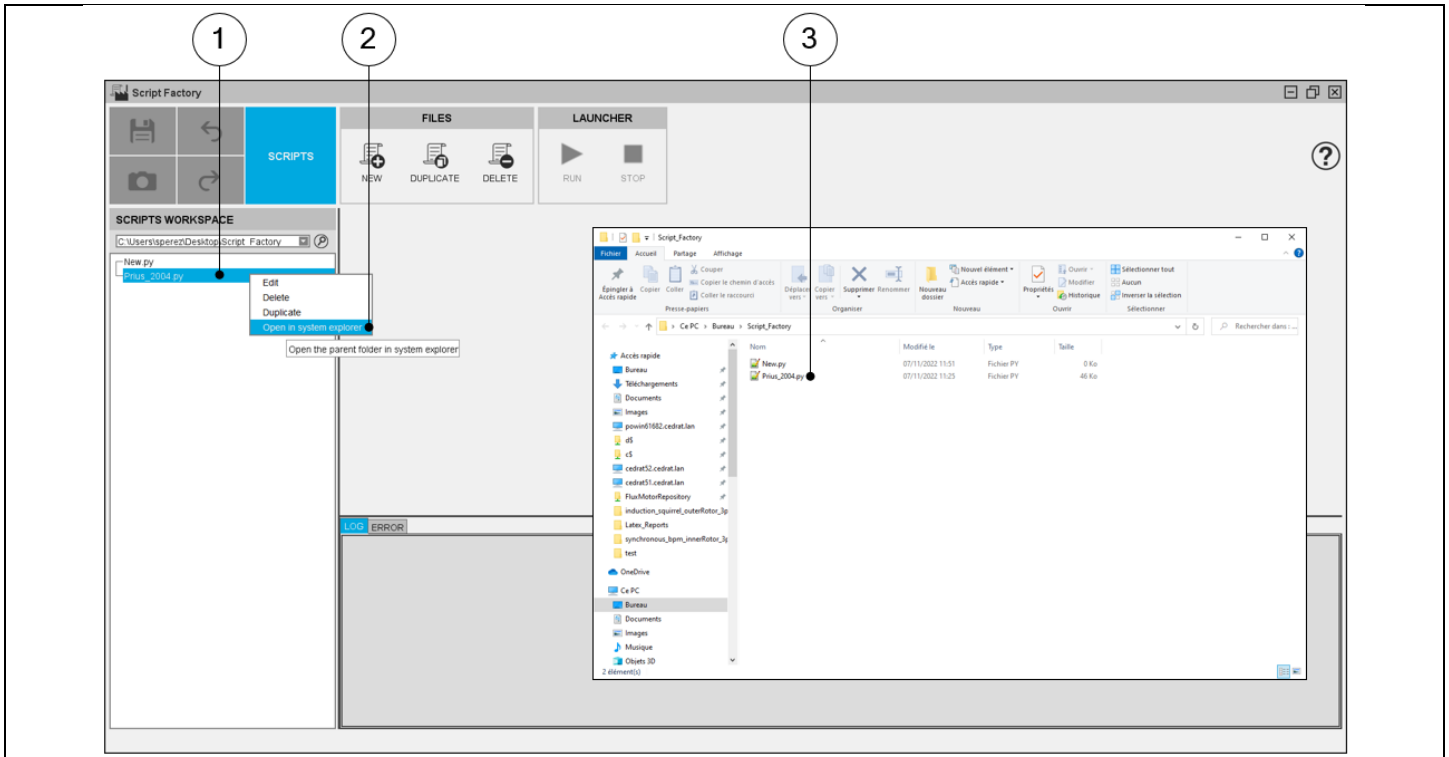


LOG and ERROR tabs

1	LOG tab gives the information for each command executed in the software and to check whether the command has been executed correctly or not.
2	ERROR tabs give the error message (as popup does in GUI environments)

## 2.8.4 Open a file in system explorer

In Script Factory application, a selected python file from the workspace tree can open in system explorer.



How to open a python Script from the workspace tree in system explorer?

1	Select a python file in the workspace tree.
2	Right click on the selected python file and select "Open in the system explorer" option.
3	The system explorer window of the selected file is opened.

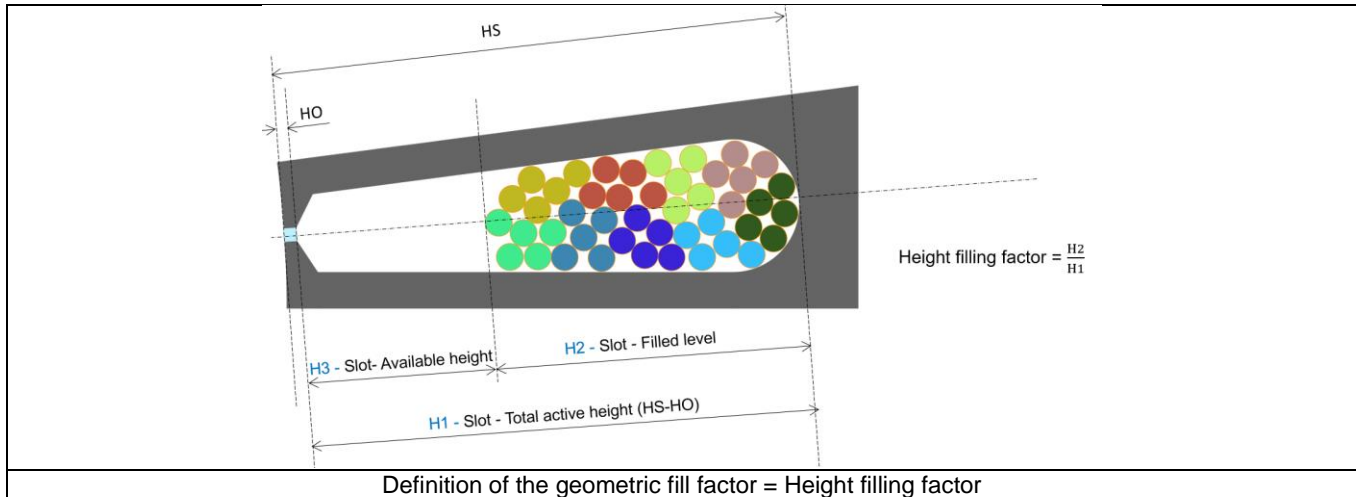
### 3 MISCELLANEOUS TOPICS

#### 3.1 Script for filling the slots

From the drop-down menu available on the left top part of Motor Factory, it is possible to open a “Debug” dialog box, in which the script commands can be written and executed.

A new command dedicated to the slot filling has been implemented. It allows to define and apply the geometric slot filling (height filling factor).

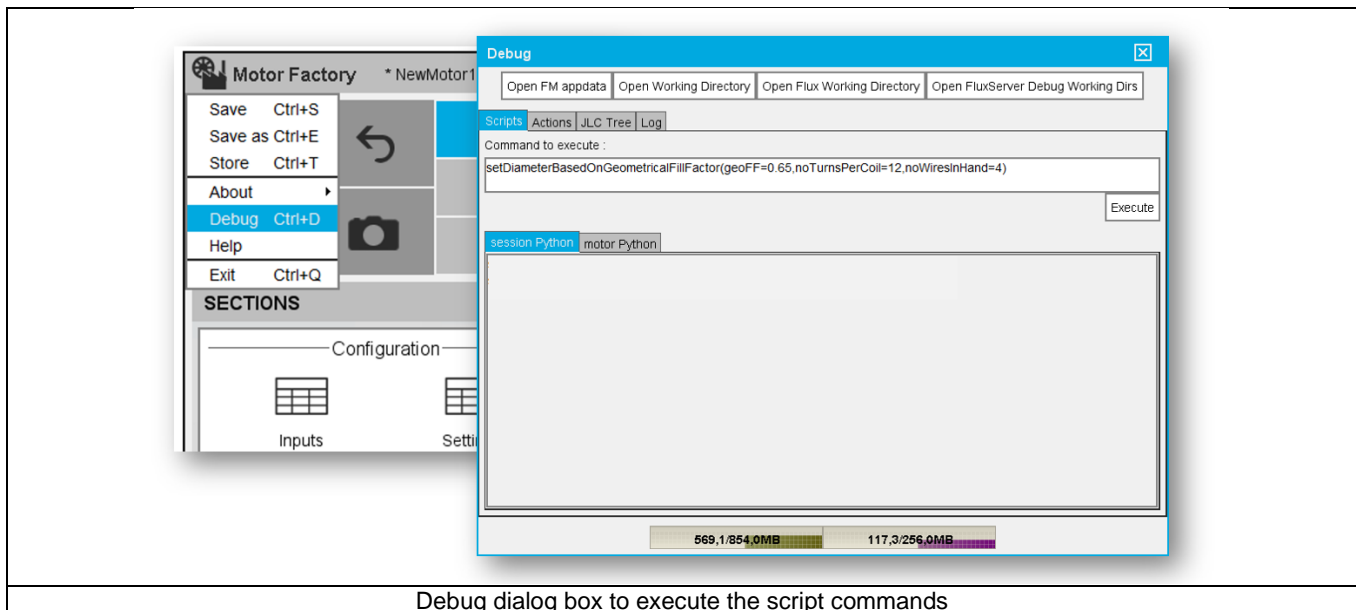
Here is the definition, Geometric fill factor = Height filling factor

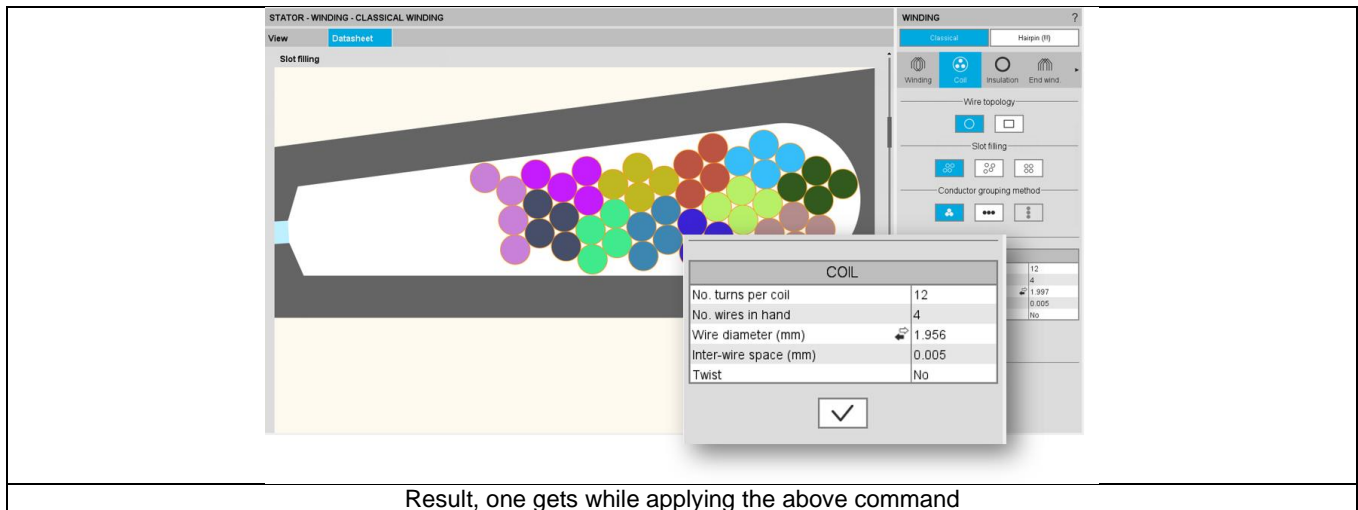


The script command is:

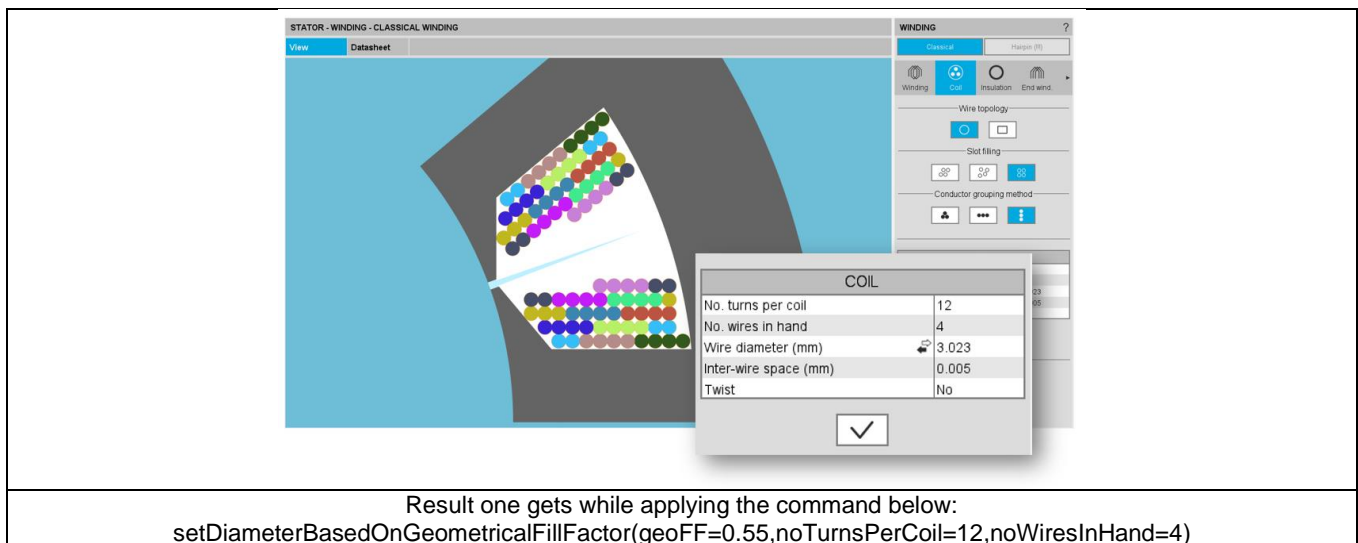
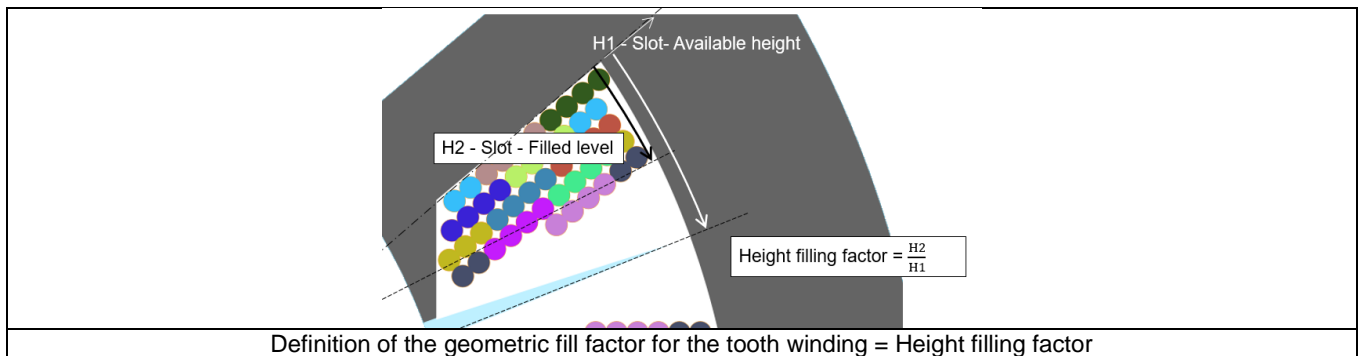
```
setDiameterBasedOnGeometricalFillFactor(geoFF=0.65,noTurnsPerCoil=12,noWiresInHand=4)
```

In this example, the goal is to find the wire diameter, which allows to define a geometric fill factor equal to 0.65 by considering a coil built with 12 turns and 4 wires in hand (in parallel).





Note: This script command can run for tooth winding slot as well. In that case the definition of the geometric fill factor = Height filling factor, can be illustrated as below:



### 3.2 Feasibility of the winding architecture and the slot filling

For information, a script command (**isWindingValid()**) allows to know if the winding is feasible or not.

This is useful from the winding architecture point of view and for the slot filling as well. This command will be able to be used inside an optimization process to select only the relevant winding configurations for example.