

POWERFACTORY

Scripting in *PowerFactory* with Python

S2021.03.04.Online_Pyt.En

March 04th - March 05th 2021

Online training course

Python is a programming language, which can be used to control *PowerFactory*. It is commonly used to automate the execution of time-consuming simulations - however, its application extends far beyond that. Python may also be used to process results, or to implement a routine that applies sequential changes to a network and calls *PowerFactory*'s analysis functions in each step.

This training course provides a compact and efficient introduction to the fundamental aspects of writing scripts inside of *PowerFactory* by using Python. The course includes basic concepts, syntax, accessing and modification of objects from within the code, automation of a series of calculations and presentation of the results etc.

Within the context of the training course numerous scripts will be created, which are supposed to encourage own ideas or which can be adapted to suit particular requirements.

WHO SHOULD ATTEND:

The training course is intended for

- Utility engineers
- Power system operators
- Project developers
- Manufacturers
- Consultants
- Electrical engineers in general

Participants should be familiar with the basic handling of DIGSILENT *PowerFactory*. Experiences with *PowerFactory*'s DPL, Python itself, as well as C++, C, Java or other scripting languages is not needed but it is an advantage.

PRICE PER PARTICIPANT:

- 984.00€* (with valid maintenance contract)
- 1,120.00€* (without valid maintenance contract)
- 336.00€* (with valid student identification)

*Prices are exclusive of VAT

Training schedule

Central European Time (UTC +01:00)

DAY 1

9:00 Fundamentals

Familiarisation with the general handling of the Python Programming Language in *PowerFactory*, e.g.: Creation of a Python script in *PowerFactory* and access to data by using Python. Presentation of the *PowerFactory* module and comparison between DPL and Python.

10:00 *PowerFactory* Object Access with Python

Access to calculation relevant objects of different classes inside of the *PowerFactory* database. Reading objects and their attributes and making necessary changes of these objects.

10:30 Coffee break

11:00 Exercise: Object Access with Python

Accessing all lines available in the network. Reading their lengths and calculating the complete length. Distinguishing whether a line is an overhead line or a cable. Presentation of loops and statements, lists and indexing in Python.

11:30 Execution of *PowerFactory* Commands with Python

Access and execution of any type of calculation objects available in *PowerFactory*.

12:30 Lunch break

13:30 Exercise: Execution of Calculation Commands

Automatic execution of load Flow calculations. Reading and reporting of calculation results in the output window.

14:30 Navigation through the *PowerFactory* Project

Navigation through the project contents and the data base.

15:00 Coffee break

15:30 Exercise: Navigation through the Project

Automatic execution of load flow calculations for multiple Study Cases and Operation Scenarios. Comparison of results and their output into the output window. Working with "dictionaries" in Python.

17:00 End of the first day

DAY 2

9:00 Report Results, Subroutines and Module

Output of customised results in the output window. Introduction of subroutines and modules. Presentation of the built-in modules SYS and IMP.

9:30 Exercise: Report Results

Access to project and load flow calculation. Reporting, whether a load flow calculation has been executed successfully.

10:30 Coffee break

11:00 Exercise: Subroutines and Module

Creating subroutines and importing them in the main script. Creating readable reports and preparing them for the export. Introduction of the lambda function and some advanced Python topics.

12:30 Lunch break

13:30 Result Files

Becoming acquainted with the Result File element (ElmRes) of *PowerFactory*. Writing/ reading data to/from it.

13:45 Exercise: Result Files

Executing multiple short circuit calculations and saving the results in a result file. Reading and reporting the previously written results.

14:30 Graphical Representation

Familiarisation with graphical representations in *PowerFactory*. Automatic creation of plots.

15:00 Coffee break

15:30 AddOn Module, Creating Objects and User Communication

Defining user specific variables via script in *PowerFactory*. Creating new objects. Make a script easy to operate for other users.

16:00 Additional Exercises:

- Engine Mode: Starting *PowerFactory* from the Python interpreter
- Exporting/importing data to/from external files
- Advanced Python Topics and Discussion: Creation of a graphical interface

17:00 End of the training course