



POWERFACTORY

Programming in PowerFactory (Python)

Course Content

DlgSILENT Buyisa (Pty) Ltd

POWER SYSTEM SOLUTIONS
MADE IN GERMANY

Programming in PowerFactory (Python)

3 Day Course

Objective:

To Enable advanced users to create Python scripts in PowerFactory for automation and advanced use of PowerFactory.

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.

Pre-requisites:

- **MUST have attended the PowerFactory Basic Course**
- Should have a good working knowledge of Python.

No of participants:

In-house at Customer premises: Minimum: 6; Maximum: 12.

At DigSILENT Buyisa Training Centre: Minimum: 10; Maximum: 16.

Online: Minimum 6; Maximum: 16.

ECSA CPD Accredited and Points:

- The course is fully accredited with the Engineering Council of South Africa (ECSA).
- 3 CPD points for completion.



POWERFACTORY

Who Should Attend:

The course is intended for:

- Utility engineers
- Power system operators
- Project Developers
- Manufacturers
- Consultants and electrical engineers

Participants should be familiar with the basic handling of DIgSILENT PowerFactory. Experience with PowerFactory DPL, Python itself, as well as C++, C, Java or other scripting languages is not needed but is an advantage.

Price per participant:

- For course pricing, kindly visit our website at: <https://digsilent.co.za/training-courses/>
 - For in house prices at customer premises: contact DIgSILENT for a quote via email info@digsilent.co.za or Telephonically (+27) 087 351 6159.
- ❖ Prices are exclusive of VAT
 - ❖ Please note that cost excludes your Company's internal administrative costs.
 - ❖ All prices may change without prior notice - please contact DIgSILENT Buyisa for the latest prices before booking.
 - ❖ **DISCOUNT** is offered if a company sends more than one delegate per course.
 - ❖ Trainings held at DIgSILENT Buyisa Training Centre includes light breakfast snack, lunch and refreshments.

Training schedule

DAY 1

08:30 Fundamentals Python Scripting in PowerFactory

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. Presentation of loops, statements, lists and indexing in Python.

Exercise: Hello PowerFactory

Creating a Python script command (ComPython) in *PowerFactory* and displaying different messages in the output window.

10:30 Tea/Coffee break

11:00 PowerFactory Objects access with Python

Access to calculation relevant objects of different classes inside of the *PowerFactory* database. Read and modify attributes of objects.

Exercise: Object access with Python

Accessing all elements of a specific class in the network. Reading their attributes and working with the values. Using attributes to categorise elements into different groups and modifying attribute values.

Q&A Session

12:30 Lunch break

13:30 Execution of PowerFactory commands with Python

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

Exercise: Execution of calculations commands with Python

Automatic execution of the Load Flow Calculation command, while adapting settings in the command. Reading of calculation results from network elements.

15:00 Tea/Coffee break



POWERFACTORY

15:30 **Navigation through the PowerFactory Project**
Showcase of different methods for accessing objects in *PowerFactory*.
Navigation through the project contents and the database.

Exercise: Navigation through the PowerFactory project

Applying different methods to access relevant objects in *PowerFactory*.
Automatic execution of load flow calculations for multiple study cases.
Checking for valid calculation results and reporting critical values.

Q&A Session

16:30 **End of the first day**

DAY 2

08:30 **Python Functions and Remote Scripts**
Introduction of functions in Python. Use of *PowerFactory* methods to obtain descriptions and units for *PowerFactory* attributes. Use of input parameters and remote scripts.

Exercise: Reporting Results

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

10:30 **Tea/Coffee break**

11:00 **Create, Delete and Connect Network Elements**
Introduction of the methods to modify a network model. Working with characteristics.

Exercise: Network modifications

Creating a new load in a network model and connecting it. Assigning time characteristics to network elements and executing a Quasi-Dynamic Simulation.

12:30 **Lunch break**



POWERFACTORY

13:30 Results Files

Familiarisation with the Result File element (ElmRes) in PowerFactory. Read and write a Result File and export its data.

Exercise: Result Files

Reading data with different methods from an existing Result File and analysing the findings. Exporting results into a csv-file.

15:00 Coffee break

15:30 Graphical Representation of Results

Familiarisation with the anatomy of plots in PowerFactory. Automatic creation of plots.

Exercise: Plotting of calculation results

Creating plots in PowerFactory via script.

16:30 End of the second day

DAY 3

08:30 Performance

Introduction to the Environment Functions in PowerFactory and best practises for efficient scripting with Python in PowerFactory.

09:00 Import and Export of Data

Introduction of possibilities to import and export data to the PowerFactory database via script.

Exercise: Import and export projects

Preparing a Python script to automatically import a project file into the database and one script for exporting a project to a file.

10:30 Tea/Coffee break



POWERFACTORY

- 11:00 Engine Mode**
Start of PowerFactory from a Python interpreter.
- Exercise: Engine Mode**
Working in GUI-less unattended mode.
- 12:30 Lunch break**
- 13:30 Parallelisation**
Options for parallel computation and setting up of the Task Automation command via script.
- 14:00 User Interaction**
Introduction of different methods for user interaction. Integration of scripts in the PowerFactory GUI via User-defined Tools.
- Exercise: User Interaction**
Implementing user input parameters and verifying the entries. Offering selection browsers and opening command windows for user input during the script execution.
- 15:00 Coffee break**
- 15:30 Add-on Module**
Defining user specific variables via script in *PowerFactory*.
- Exercise: AddOn Module**
Creating AddOn attributes for a PowerFactory object class and writing values onto these attributes.
- Q&A Session**
- 16:30 End of the third day**