



# PowerWorkshop Professional Python for Embedded

Within the embedded computing sphere, the programming language C has long been considered the standard. However, more complex applications and faster time-to-market requirements call for alternatives. Traditionally used for web and desktop applications, Python offers such an alternative, thanks to its support for C/C++ libraries and the provision of frameworks for processing complex algorithms. Those associated frameworks enable the development of complex algorithms used across data analytics, machine learning (ML), and artificial intelligence (AI) applications. Of course, these applications are hot topics within embedded computing and are driving the adoption of Python, especially within Industrial IoT (IIoT) at the edge.

XILINX offers an open-source framework (PYNQ) that allows Python to interact with the Zynq portfolio. This is realized by hybrid libraries, which is a new form of libraries. Such a hybrid library includes both a bitstream and the associated hardware-related C code, but also a programming interface (API) for Python. In this course you

will learn the basics of programming with Python, which you will use in a workshop. You will then be able to develop prototypes based on Python on embedded systems in a short time.

As a participant of the extended workshop, you will also receive a profound training in the use of libraries to process large data, which will form the basis for application development in the field of AI, ML and Big Data.

## Applicable technologies

Python, PYNQ

## Requirements

Basic knowledge on OOP

## Duration and Costs:

Duration: 5 days

Cost: € 3.100 net per person, including detailed training material, drinks in the breaks and lunch.

## Agenda

### Introduction

#### First Steps

- Environment Setup
- Reasons to use Python
- The use of Jupyter

### Development with Python

#### Basics

- Variables
- Types, I/O and Import
- Operators & Namespace
- Datatypes

#### Flow Control

- Statements
- Loops

#### Functions

- Function & Argument
- Recursion
- Modules & Packages

#### Objects and Classes

- OOP
- Inheritance

### Advanced

- Iterators
- Generators
- Dekorators
- Closures

### Data Analysis with Python

- Structures & Sorting
- Operators & Algorithms
- Statistics
- "Wrangling"
- Visualization

### Embedded Hardware

#### Basics

- PYNQ Architecture
- Board Setup

#### Libraries

- CTypes

#### Workshop - Applications

- GPIO App
- Face & Eye Detection
- HDMI Streaming & Processing

#### FPGA

- Creation of a custom overlay (Vivado)