

## Fundamentals of Python

**Level:** foundation

**Length:** 35 – 40 hours

**Course Objective:** solid introduction to Python programming language, mainly its support for procedural and object oriented programming, how it is used in order to solve problems.

### What You Will Learn

- How the code is organized, software development cycle
- Language elements: data structures, control structures, basic types, functions
- Python's support for object oriented programming (OOP)
- Exercise programming with Python
- Exercise the soft skills of communication, team work, presentation of ideas and solutions

**Who can participate:** anybody who wants to learn Python

### Prerequisites

- Basic notions of programming: familiar with the notion of data type, variable, flow control, ability to describe an algorithm
- Comfortable with using the host operating system
- The knowledge of another programming language, for example C, Perl, C++, Java, etc.) would be helpful to easier learn Python

**Required infrastructure:** VGA projector, whiteboard, workstation with Python installed and an IDE for writing code, for example Community PyCharm

### Bibliography

- Programming in Python 3. A Complete Introduction to the Python Language, Mark Summerfield, Second Edition, Addison-Wesley, 2010, ISBN-10 0-321-68056-1
- Beginning Python: From Novice to Professional, Second Edition, Magnus Lie Hetland, Apress, 2008, ISBN-10 1-59059-982-9
- The Quick Python Book, Second Edition, Vernon L. Ceder, Manning, 2010, ISBN 9781935182207

**Related Courses:** Advanced Python Programming

## Description

This is a fundamental training for any programmer who wants to use Python regardless the context – application scale, industry, system purpose. It contains the mandatory language elements and constructs used by any Python program.

The course is designed for programmers who want a solid introduction to Python, the focus is the language, its elements – data structures, control structures, how the programs are organized, particularities of using it in procedural programming, object oriented programming and even functional programming.

Python is exercised by examples and practical assignments, it is shown how the problems are approached and solved, particularities and stereotypes of using this language.

Based on this course the learning of Python might continue with more advanced constructs of the language (see Advanced Python Programming), or specialized fields of application of Python, for example: testing with Python, automation with Python, applications related to machine learning, deep learning, big data, etc.

## Contents

1. Introduction to Python. The place of Python among other programming languages. Main characteristics, code life cycle, Python distributions.
2. Building and running Python programs. Quick introduction to basic language constructs used to building the programs.
3. Predefined data types for representing common entities like numbers and strings. Identifiers and operators.
4. Collections. The three main categories, main operations, how are they used.
5. Control structures and functions. How to function work and how they are used. Lambda expressions. Capture of variables from context, exercise them.
6. Exceptions – mechanism, context of using, particularities.
7. Modules and packages. General introduction to Python standard library. Installing third party libraries.
8. Classes and objects, basic language support for object oriented programming: inheritance & delegation, polymorphism, proprieties.
9. Working with text & binary files. Standard serialization with pickles.
10. Debugging, testing and profiling Python programs.