

Fundamentals of Python

Level: beginner

Length: 35 hours

Course Objective: solid introduction to Python programming language, mainly its object oriented support and particularities of using it.

What You Will Learn

- How the code is organized, software development cycle
- Language elements: data structures, control structures, basic types
- Python's support for object oriented programming (OOP)
- Exercise OOP 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
- Comfortable with using the host operating system
- The knowledge of another programming language, for example C, Perl, C++, Java, etc.) would be helpful to learn easier Python

Required infrastructure: VGA projector, whiteboard, workstation

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 Topics

Description

The course is designed for programmers who want a quick and solid introduction to Python. The focus is the language, its constructive 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.

Contents

1. Introduction to Python. Building and running Python programs. Quick introduction to basic language constructs used in construction the programs.
2. Data types. Identificators and operators. Integer types. Floating point types. Strings.
3. Collections data types. Sequences. Sets. Dictionaries. Iteration and copying collections.
4. Control structures and functions. Exception handling. Functions.
5. Modules and packages. General introduction to Python standard library.
6. Classes and objects, language support for object oriented programming.
7. Working with files.
8. Debugging, testing and profiling Python programs.