Python Foundation Course

A hands on course for mastering the essentials of Python programming language and using it to solve real-world problems.

## Getting Started

Quick introduction to Python programming language, including basic data types, functions, methods, modules, conditionals and loops.

## Working with Data

Introduction to Python’s powerful data structures including, lists and dictionaries. Also covers list comprehensions, processing text and working with files.

## Modules

Introduction to Python module system, importing modules, writing custom modules, documenting code using docstrings and installing third-party modules. This section concludes with a tour of Python standard library covering modules related to file system handling, downloading stuff from web, JSON and working with APIs.

## Classes and Objects

Introduction to object-oriented programming with Python. Describes how classes offer a different programming model. Covers writing classes, object creation, inheritance and exception handling.

## Testing Python Programs

Introduction to automated testing using unittest and py.test modules.

Advanced Python

An intensive hands-on course that dives deep into the Python internals, advanced features like decorators, generators, meta classes etc. and best practices of Python programming language.

## Warm up

Review of Python programming language. Topics covered are lists, dictionaries, list comprehensions and importing modules.

## Higher ­Order Functions and Decorators

A deeper look at functions in Python. Covers functions with variable arguments, keyword and default arguments, scoping rules, recursion, higher order functions and decorators.

## Iterators and Generators

Introduction to Iterators, generators and generator expressions with emphasis on working with large data and how these techniques help to make code more readable. Also explores coroutines using the new async and await syntax and async programming.

## Deeper look at classes and objects

Understanding classes and objects at a deeper level. Covers old-style and new-style classes, static methods, class methods, special methods for operator overloading, slots, descriptors, context managers and meta classes.

## Writing Python Libraries

Covers best practices of writing, documenting, testing and distributing python libraries.