

## Python 3 Essentials

### Overview

---

An introductory and beyond-level practical, hands-on Python training course that leads the student from the basics of writing and running Python scripts to more advanced features.

### Target Audience

---

Experienced Programmers and Systems Administrators.

### Course Objectives

---

Throughout the course students will be led through a series of progressively advanced topics, where each topic consists of lecture, group discussion, comprehensive hands-on lab exercises, and lab review. This course is "skills-centric", designed to train attendees in core Python and web development skills beyond an intermediate level, coupling the most current, effective techniques with best practices. Working within an engaging, hands-on learning environment, guided by our expert Python practitioner, students will learn to:

- Create working Python scripts following best practices
- Use python data types appropriately
- Read and write files with both text and binary data
- Search and replace text with regular expressions
- Get familiar with the standard library and its work-saving modules
- Use lesser-known but powerful Python data types
- Create "real-world", professional Python applications
- Work with dates, times, and calendars
- Know when to use collections such as lists, dictionaries, and sets
- Understand Pythonic features such as comprehensions and iterators
- Write robust code using exception handling

### Course Outline

---

#### 1 - An Overview of Python

What is python?

1 -- An overview of Python

What is python?

Python Timeline

Advantages/Disadvantages of Python

Getting help with pydoc

## 2 - The Python Environment

- Starting Python
- Using the interpreter
- Running a Python script
- Python scripts on Unix/Windows
- Editors and IDEs

## 3 - Getting Started

- Using variables
- Built-in functions
- Strings
- Numbers
- Converting among types
- Writing to the screen
- Command line parameters

## 4 - Flow Control

- About flow control
- White space
- Conditional expressions
- Relational and Boolean operators
- While loops
- Alternate loop exits

## 5 - Sequences

- About sequences
- Lists and list methods
- Tuples
- Indexing and slicing
- Iterating through a sequence
- Sequence functions, keywords, and operators
- List comprehensions
- Generator Expressions
- Nested sequences

## 6 - Working with files

- File overview
- Opening a text file
- Reading a text file
- Writing to a text file
- Reading and writing raw (binary) data
- Converting binary data with struct

## 7 - Dictionaries and Sets

- About dictionaries
- Creating dictionaries
- Iterating through a dictionary
- About sets
- Creating sets
- Working with sets

## 8 - Functions

- Defining functions
- Parameters
- Global and local scope
- Nested functions
- Returning values

## 9 - Sorting

- The sorted() function
- Alternate keys
- Lambda functions
- Sorting collections
- Using operator.itemgetter()
- Reverse sorting

## 10 - Errors and Exception Handling

- Syntax errors
- Exceptions
- Using try/catch/else/finally
- Handling multiple exceptions
- Ignoring exceptions

## 11 - Modules and Packages

- The import statement
- Module search path
- Creating Modules
- Using packages
- Function and Module aliases

## 12 - Classes

- About o-o programming
- Defining classes
- Constructors
- Methods
- Instance data
- Properties
- Class methods and data

## 13 - Regular Expressions

- RE syntax overview
- RE Objects
- Searching and matching
- Compilation flags
- Groups and special groups
- Replacing text
- Splitting strings

## 14 - The standard library

- The sys module
- Launching external programs
- Math functions
- Random numbers
- The string module
- Reading CSV data

## 15 - Dates and times

- Working with dates and times
- Translating timestamps
- Parsing dates from text
- Formatting dates
- Calendar data

## 16 - Working with the file system

- Paths, directories, and filenames
- Checking for existence
- Permissions and other file attributes
- Walking directory trees
- Creating filters with fileinput
- Using shutil for file operations
- 17 – Advanced data handling
- Defaultdict and Counter
- Prettyprinting data structures
- Compressed archives (zip, gzip, tar, etc.)
- Persistent data

## 17 - Advanced data handling

- Defaultdict and Counter
- Prettyprinting data structures
- Compressed archives (zip, gzip, tar, etc.)
- Persistent data

## 18 - Network services

- Grabbing web content
- Sending email
- Using SSH for remote access
- Using FTP

## 19 - Writing real-life applications

- Parsing command-line options
- Detecting the current platform
- Trapping signals
- Implementing logging
- Python Timeline
- Advantages/Disadvantages of Python
- Getting help with pydoc

---