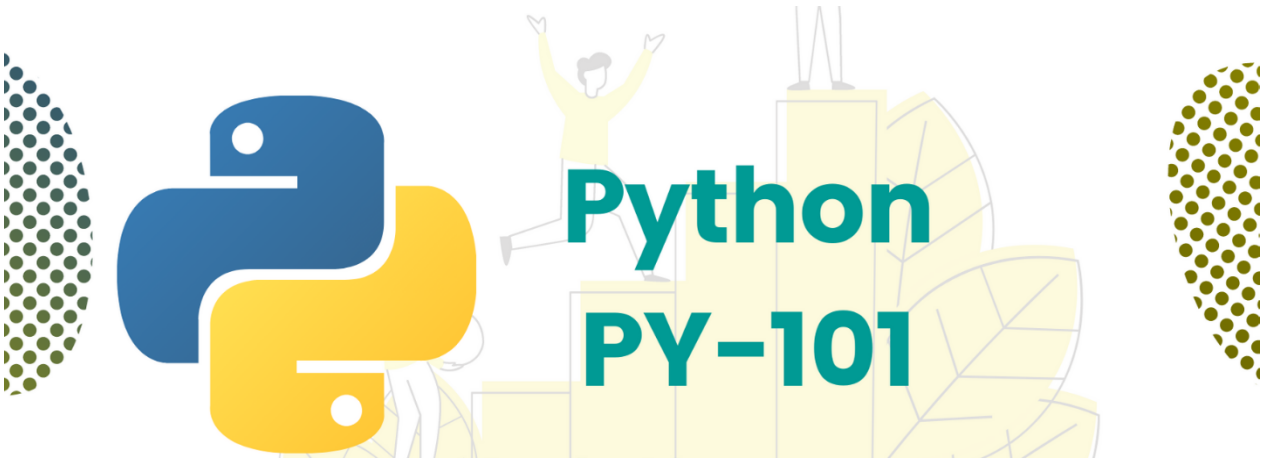

One Week Python Program

Course Number: 2020-PY-101 **Duration:** 5 Days

Overview

This Python for beginners training course leads the students from the basics of writing and running Python scripts to more advanced features such as file operations, working with binary data, and using the extensive functionality of Python modules. Extra emphasis is placed on features unique to Python, such as tuples, array slices, and output formatting.

Prerequisites

Students should already be comfortable using the operating system (Linux, Unix, Windows, Solaris, Mac OS X, etc.) on which they will be running Python. While not mandatory, basic skills with at least one other programming language are desirable.

Materials

All Python training students will receive comprehensive courseware in GitHub.

Software Needed on Each Student PC

- Any Windows, Linux, or Mac OS X operating system
- Python 3 or later
- An IDE with Python support (PyCharm Community Edition is an excellent free option, but there are several other good ones)

Objectives

- Master the fundamentals of writing Python scripts
- Learn core Python scripting elements such as variables and flow control structures
- Discover how to work with lists and sequence data
- Write Python functions to facilitate code reuse
- Use Python to read and write files
- Make their code robust by handling errors and exceptions properly
- Work with the Python standard library
- Explore Python's object-oriented features
- Search text using regular expressions

Course Schedule

Week	Subject	Practice Problems
Day-1	An Overview of Python The Python Environment Getting Started Operations	Task-1-5
Day- 2	Flow Control Data Structure	Task-5-10
Day-3	Working with Files Function Sorting	Task 11-15
Day-4	Errors and Exception Handling Modules and Packages	Task 16-20
Day-5	Highlights of the Standard Library An Introduction to Python Classes	Task 20-25

Outline

- An Overview of Python
  - What is Python?
  - Interpreted languages
  - Advantages and disadvantages
  - Downloading and installing
  - Which version of Python
  - Where to find documentation
- The Python Environment
  - Structure of a Python script
  - Using the interpreter interactively
  - Running standalone scripts under Unix and Windows
- Getting Started
  - Using variables
  - String types: normal, raw and Unicode
  - String operators and expressions
  - Math operators and expressions
  - Writing to the screen
  - Command line parameters
  - Reading from the keyboard
- Flow Control
  - About flow control
  - Indenting is significant
  - The if and elif statements
  - while loops

    - Using lists
    - Using the for statement
    - The range() function
- Data Structure
    - list operations
    - list methods
    - Strings are special kinds of lists
    - Tuples
    - Dictionary
    - Set
    - Dictionary overview
    - Creating dictionaries
    - Dictionary functions
    - Fetching keys or values
    - Testing for existence of elements
    - Deleting elements
    - Sets And Frozen Sets
    -
- Working with Files
    - Text file I/O overview
    - Opening a text file
    - Reading text files
    - Raw (binary) data
    - Writing to a text file
- Functions
    - Syntax of function definition
    - Formal parameters
    - Global versus local variables
    - Passing parameters and returning values
- Sorting
    - The sorted() function
    - Alternate keys
    - Multiple keys
    - Lambda functions
- Errors and Exception Handling
    - Dealing with syntax errors
    - Exceptions
    - Handling exceptions with try/except
    - Cleaning up with finally
- Modules and Packages
    - What is a module?
    - The import statement
    - Function aliases
    - Packages
- Highlights of the Standard Library
    - Working with the operating system
    - Grabbing web pages
    - Sending email
    - math and random
    - Accessing dates and times with datetime
    - Working with compressed files
- An Introduction to Python Classes
    - About o-o programming

    - Defining classes
    - Constructors
    - Instance methods
    - Instance data
    - Class methods and data
    - Destructors
- Conclusion