

# **Python Course Syllabus**

Duration: **30hrs**  
Mode: **Class room/Online**

## **1. Introduction**

- 1.1. History of Python
- 1.2. Python Features
- 1.3. Why name Python?
- 1.4. Installation in Windows and Linux
- 1.5. Why do we need Python?
- 1.6. Difference between Programming language vs Scripting language
- 1.7. Difference between Interpreter vs Compiler
- 1.8. What is object oriented Python
- 1.9. Coding style
- 1.10. Indentation
- 1.11. Running python
- 1.12. How python differ from other programming language?
- 1.13. Difference between Python 2 and Python 3
  - 1.13.1. Syntax different between 2 and 3
- 1.14. Python symbols and meanings

## **2. Execution steps**

- 2.1. Interactive Shell
- 2.2. Executable or script files
- 2.3. User Interface or IDE

## **3. Memory management and Garbage collections**

- 3.1. Variable creation
  - 3.1.1. Rules for creation of variables
- 3.2. Local
- 3.3. Global
- 3.4. Object creation and deletion
- 3.5. Object properties

## **4. Data Types and Operations**

- 4.1. Numbers
- 4.2. Strings
  - 4.2.1. Formatting in string
  - 4.2.2. String extractions and operations
- 4.3. List
  - 4.3.1. List manipulations
  - 4.3.2. List operation
  - 4.3.3. Set function
- 4.4. Tuple
  - 4.4.1. Tuple manipulation
  - 4.4.2. Tuple vs List
- 4.5. Dictionary
  - 4.5.1. Multidimensional dictionary
- 4.6. Id and Values
- 4.7. Zip function

## **5. Split and Join**

## **6. Shallow vs deep copy**

## **7. Python operators**

- 7.1. Operators name and usage
- 7.2. Expressions in print

## **8. User inputs**

- 8.1. Getting user inputs
- 8.2. Difference between input functions in 2.x and 3.x
- 8.3. Evaluating the user inputs

## 9. Unpacking command line arguments

- 9.1. Understanding external modules
- 9.2. Getting command line arguments
- 9.3. Understanding the command line arguments

## 10. Decision Statements and Syntax

- 10.1. If tests and Syntax Rules
- 10.2. If .. elif .. else
- 10.3. Nested conditional statements
- 10.4. Using logical operators
- 10.5. IF statement in one line
- 10.6. Ternary operator
- 10.7. Data type compression

## 11. Loops

- 11.1. While loops
  - 11.1.1. Infinite loop
- 11.2. For Loops
  - 11.2.1. For loop in all Data types
- 11.3. Nested loops
- 11.4. Loop control statements

## 12. Range vs xrange

## 13. Functions

- 13.1. Function definition
- 13.2. Empty function creation and calling it
- 13.3. Call by value vs call by reference vs call by object
- 13.4. Calling a function with different set of arguments
  - 13.4.1. Without arguments
  - 13.4.2. Required arguments
  - 13.4.3. keyword arguments
  - 13.4.4. Default arguments
  - 13.4.5. Required and keyword arguments
  - 13.4.6. Arbitrary arguments
  - 13.4.7. Required, keyword and arbitrary arguments
  - 13.4.8. Arbitrary keyword arguments
  - 13.4.9. Required, keyword, arbitrary and keyword arbitrary arguments Function Scope

## 14. File I/O Operations

- 14.1. Opening and closing a file
- 14.2. Reading a file
- 14.3. Writing a file
- 14.4. Appending a file
- 14.5. Using Files
- 14.6. Other File tools
- 14.7. Os module functions

## 15. Exception Handling

- 15.1. Default Exception Handler
- 15.2. Catching Exceptions
- 15.3. Raise an exception
- 15.4. User defined exception

## 16. Variable scope

- 16.1. Local vs Global

## 17. Regular Expression

- 17.1. What is regular expression
- 17.2. Why we need to learn regular expression
- 17.3. Importance of regular expression
- 17.4. Re module usage
- 17.5. Match function
- 17.6. Search and replace
- 17.7. match() vs search()

- 17.8. Regular expression modifier
- 17.9. Regular expression patterns
  - 17.9.1.Repetition character classes
  - 17.9.2.Regular expression modes
  - 17.9.3.Compiling regular expression
  - 17.9.4.Escaping character
  - 17.9.5.More Pattern Power
  - 17.9.6.Grouping the patterns
  - 17.9.7.Modifying Strings
  - 17.9.8.Greedy versus Non-Greedy
  - 17.9.9.Finding all occurrences
  - 17.9.10.Search and replace
  - 17.9.11.And much more advanced regular expressions

## 18.Lambda function

- 18.1. Lambda in map
- 18.2. Lambda in filter
- 18.3. Lambda in reduce

## 19.List comprehension

## 20.Generator

## 21.Generator comprehension

## 22.Iterator

## 23.Dictionary comprehension

## 24.Modules

- 24.1. Understanding and creation of module
- 24.2. User defined module and importing module
- 24.3. Types of importing module and module functions
- 24.4. Using external module
- 24.5. Checking existence of module
- 24.6. Understanding the default module path and module search hierarchy
- 24.7. Module just once
- 24.8. Invoking Linux commands thru external module
- 24.9. Invoking another script in python
- 24.10.Linux commands in python module

## 25.Packages

- 25.1. Importance of package
- 25.2. Package creation
- 25.3. Calling package

## 26.Classes

- 26.1.What is class?
- 26.2. Empty class creation
- 26.3. Attribute vs instance reference
- 26.4. Classes method calls
- 26.5. Function vs Method
- 26.6. Object creation and accessing attributes
- 26.7. Static and Class Methods