



Total Course Duration: 50 hours
8 Weeks: Weekend Class (Sat and Sun)

Python is an easy to learn, powerful programming language. It has efficient high-level data structures and a simple but effective approach to object-oriented programming. Python's elegant syntax and dynamic typing, together with its interpreted nature, make it an ideal language for scripting and rapid application development in many areas on most platforms.

Course Content:

1. Introduction to Python:

- What is Python?
- Why Python?
- Python Applications in real life
- Brief history of Python
- Versions of Python
- Installing Python
- Using IDLE
- First Python Program
- Getting help from Python Docs

2. Variables Data types

- Intro to dynamic typing
- Variables in Python
- Naming conventions
- Basic Data types (representation of strings, integer, floats)

3. Basic Syntax

- Basic syntax
- Commenting
- Indentation
- Python keywords
- Strings
- String values

- String Operations
- String slicing
- Built in string methods
- Formatted printing
- Simple Input and Output handling

4. Language Building blocks

- Control statements, the if, elif, else
- True and False
- Arithmetic Operators
- Relational Operators
- Logical Operators
- Bitwise Operators
- While loop
- Usage of pass, break and continue
- For each loop

5. Collections

- Lists
- Tuples
- Sets
- Dictionaries
- Sorting collections
- Operations on collections
- Discussion on real life application of above collections

6. Functions

- Introduction to functions
- Built in functions
- User defined functions
- Function parameters
- Variable arguments ,args and kwargs
- Positional and named arguments
- Discussion scope of variables with respect to functions and namespace
- Passing function to another function

7. Project I

8. File Handling

9. Modules

- Introduction to modules
- Introduction to standard modules
- **OS** module
- **path** module
- **Sys** module
- **sub process** module

- Argument parsing using *argparse* module
- .csv file parsing using *csv* module
- .json file parsing using *Jason* module
- Xml file parsing using *xml* module
- Introduction to *logging* module

10. Project 2: Building log parser and reporting the results

11. Object Oriented Programming

- Introduction to Classes and Objects
- Principles of OOP
- Instance methods
- Special methods
- Encapsulation
- Inheritance
- Polymorphism

12. Regular Expressions

- Introduction to regular exceptions
- Introduction to *re* module
- Simple character matches
- Match function
- Searching function
- Regular expression patterns
- Patterns in Regex
- Search And Replace

13. Optional I(for testers)

- Introduction to testing using Python
- Introduction to test automation
- Introduction to Selenium web driver
- Web testing using selenium

14. Option II (developers)

Advance topics:

- Generators
- Decorators
- Iterators and iterator protocol
- Debugging using PDB

15. Options III(Web programming)

- Introduction to web programming using Python
- Introduction to Django/Flask

- Introduction to Restful API's using Python

16. Option IV(Data science)

- Introduction to data science using python
- Introduction to pandas module
- Introduction to data visualization using matplotlib
- Introduction to numpy
- Introduction to scipy

17. Coding Challenges in Python

Assessment test

18. Interview Guidance

Mock Interview for Technical Interview