

# **BASIC & ADVANCE PYTHON SYLLABUS**

## **MODULE 1: Introduction to Python**

### **1.1 What is Python?**

- History of Python
- Features of Python
- Applications of Python
- Advantages and Limitations

### **1.2 Python Installation**

- Installing Python
- Setting up Python Environment
- Python IDLE
- Using Python with VS Code / PyCharm

### **1.3 First Python Program**

- Hello World Program
- Running Python Scripts
- Interactive Mode vs Script Mode

### **1.4 Python Syntax Basics**

- Indentation
- Comments
- Keywords
- Identifiers

## MODULE 2: Variables and Data Types

### 2.1 Variables

- Creating Variables
- Multiple Assignment
- Naming Rules

### 2.2 Data Types

- Numeric Types (int, float, complex)
- Boolean
- String
- NoneType

### 2.3 Type Conversion

- Implicit Conversion
- Explicit Conversion (Type Casting)

### 2.4 Input and Output

- input()
- print()
- Formatting Output

## MODULE 3: Operators in Python

### 3.1 Arithmetic Operators

- +, -, \*, /, %, //, \*\*

### 3.2 Assignment Operators

- =, +=, -=, \*=, /=

### 3.3 Comparison Operators

- ==, !=, >, <, >=, <=

### 3.4 Logical Operators

- and, or, not

### 3.5 Bitwise Operators

- &, |, ^, ~, <<, >>

### 3.6 Membership Operators

- in, not in

### 3.7 Identity Operators

- is, is not

## MODULE 4: Control Structures

### 4.1 Conditional Statements

- if
- if-else
- if-elif-else
- Nested if

### 4.2 Looping Statements

- for loop
- while loop

### 4.3 Loop Control Statements

- break
- continue
- pass

### 4.4 Nested Loops

## MODULE 5: Strings

### 5.1 String Basics

- Creating Strings
- Accessing Characters

### 5.2 String Operations

- Concatenation
- Repetition
- Membership

### 5.3 String Slicing

### 5.4 String Methods

- upper()
- lower()
- strip()
- replace()
- split()
- join()
- find()

## 5.5 String Formatting

- format()
- f-strings

## MODULE 6: Lists

### 6.1 Creating Lists

### 6.2 Accessing List Elements

### 6.3 List Operations

- append()
- extend()
- insert()
- remove()
- pop()

### 6.4 List Methods

- sort()
- reverse()
- count()
- index()

### 6.5 List Slicing

### 6.6 Nested Lists

### 6.7 List Comprehension

## **MODULE 7: Tuples**

### **7.1 Creating Tuples**

### **7.2 Accessing Tuple Elements**

### **7.3 Tuple Operations**

### **7.4 Tuple Packing and Unpacking**

### **7.5 Tuple Methods**

## **MODULE 8: Sets**

### **8.1 Creating Sets**

### **8.2 Set Operations**

- union()
- intersection()
- difference()

### **8.3 Set Methods**

- add()
- remove()
- discard()
- clear()

## **MODULE 9: Dictionaries**

### **9.1 Creating Dictionaries**

### **9.2 Accessing Values**

### 9.3 Dictionary Methods

- keys()
- values()
- items()
- update()
- pop()

### 9.4 Nested Dictionaries

### 9.5 Dictionary Comprehension

## MODULE 10: Functions

### 10.1 Defining Functions

### 10.2 Calling Functions

### 10.3 Function Arguments

- Positional Arguments
- Keyword Arguments
- Default Arguments
- Variable-length Arguments (\*args, \*\*kwargs)

### 10.4 Return Statement

### 10.5 Lambda Functions

### 10.6 Recursion

## MODULE 11: Modules and Packages

### 11.1 What are Modules?

## 11.2 Importing Modules

- import
- from import
- aliasing

## 11.3 Built-in Modules

- math
- random
- datetime
- os
- sys

## 11.4 Creating Custom Modules

## 11.5 Python Packages

## MODULE 12: File Handling

### 12.1 File Operations

- open()
- close()

### 12.2 File Modes

- r
- w
- a
- rb
- wb

## 12.3 Reading Files

- read()
- readline()
- readlines()

## 12.4 Writing Files

- write()
- writelines()

## 12.5 Working with CSV Files

## 12.6 Working with JSON Files

## MODULE 13: Exception Handling

### 13.1 Errors vs Exceptions

### 13.2 try-except Block

### 13.3 Multiple Exceptions

### 13.4 finally Block

### 13.5 Custom Exceptions

## MODULE 14: Object-Oriented Programming (OOP)

### 14.1 Classes and Objects

### 14.2 Constructor (init)

### 14.3 Instance Variables and Methods

## 14.4 Class Variables and Methods

## 14.5 Encapsulation

## 14.6 Inheritance

- Single Inheritance
- Multiple Inheritance
- Multilevel Inheritance

## 14.7 Polymorphism

## 14.8 Abstraction

## 14.9 Method Overriding

## **MODULE 15: Advanced Python Concepts**

### 15.1 Iterators

### 15.2 Generators

### 15.3 Decorators

### 15.4 Context Managers

### 15.5 Closures

### 15.6 Map, Filter, Reduce

## **MODULE 16: Working with Databases**

### 16.1 SQLite with Python

## 16.2 MySQL with Python

### 16.3 CRUD Operations

- Insert
- Select
- Update
- Delete

## **MODULE 17: GUI Programming**

### 17.1 Tkinter Basics

### 17.2 Creating Windows

### 17.3 Widgets

- Label
- Button
- Entry
- Frame
- Canvas

### 17.4 Layout Managers

- pack()
- grid()
- place()

### 17.5 Event Handling

## **MODULE 18: Web Scraping**

### 18.1 Introduction to Web Scraping

## 18.2 Requests Library

## 18.3 BeautifulSoup

## 18.4 Extracting Data from Websites

## MODULE 19: APIs and JSON

### 19.1 What is an API?

### 19.2 Making API Requests

### 19.3 Working with JSON Data

### 19.4 REST APIs with Python

## MODULE 20: Data Analysis with Python

### 20.1 NumPy Basics

### 20.2 Pandas Basics

### 20.3 Data Cleaning

### 20.4 Data Manipulation

### 20.5 Data Visualization

- Matplotlib
- Seaborn

## **MODULE 21: Machine Learning Basics**

**21.1 Introduction to Machine Learning**

**21.2 Scikit-Learn**

**21.3 Linear Regression**

**21.4 Classification Algorithms**

**21.5 Model Evaluation**

## **MODULE 22: Automation with Python**

**22.1 Automating Files**

**22.2 Sending Emails with Python**

**22.3 Working with Excel**

**22.4 Web Automation with Selenium**

## **MODULE 23: Testing in Python**

**23.1 Unit Testing**

**23.2 unittest Module**

**23.3 pytest Overview**

## **MODULE 24: Performance and Optimization**

**24.1 Profiling Code**

## 24.2 Memory Management

## 24.3 Multithreading

## 24.4 Multiprocessing

## 24.5 Async Programming

## MODULE 25: Deployment and Packaging

### 25.1 Virtual Environments

### 25.2 Packaging Python Applications

### 25.3 Creating Executable Files

### 25.4 Creating APK / Desktop Apps

## MODULE 26: Python Projects

### Beginner Projects

1. Calculator
2. Number Guessing Game
3. Password Generator
4. Contact Book
5. Quiz Game

### Intermediate Projects

6. File Organizer
7. Web Scraper
8. GUI Calculator
9. Chat Application

10. Weather App using API

### **Advanced Projects**

- 11. Student Management System
- 12. E-commerce Scraper
- 13. Machine Learning Prediction App
- 14. Desktop Notepad Application
- 15. Language Translator App