Complete Python Course Outline

Shiv_Freelancer(python.fullstack.freelancer@gmail.com)

Basics of Python Programming

Python Basics

- ✓ About Python
- ✓ input & output function
- ✓ Data Types
- ✓ Variables
- ✓ comments
- ✓ Keywords and Identifiers
- ✓ Type conversion
- ✓ Literals

Operators + conditional statements + Loops

- ✓ Operators (Arithmetic, Relational, Assignment, Membership, Identity)
- ✓ Conditional Statements (if, elif, else)
- ✓ range() function
- ✓ loops (for, while)
- ✓ Jumping statements (Break, continue, pass)
- ✓ Enumeration

Strings

- ✓ Introduction to Strings
- ✓ String indexing
- ✓ String slicing
- ✓ Edit and delete a string
- ✓ Operations on String
- ✓ Common String functions

Python Data Structures

Python Lists

- ✓ Introduction
- ✓ Array vs List
- ✓ Characteristics of Python List
- ✓ Methods:
  - Create and access a list
  - Adding item to a list (append(), extend(), insert())
  - Edit items in a list
  - Deleting items from a list
  - List comprehension
- ✓ Disadvantages of Python list

Tuples + Set + Dictionary

Tuple

- ✓ Create and access a tuple
- ✓ Can we add and modify items to a tuple?
- ✓ Operations on tuple
- ✓ Tuple functions
- ✓ List vs tuple
- ✓ Tuple packing and unpacking

Set

- ✓ Create and access a set
- ✓ Adding elements to a Set.
- ✓ Deletion
- ✓ Operations on set
- ✓ set functions
- ✓ Frozen set (immutable set)
- ✓ Set comprehension

Dictionary

- ✓ Create dictionary
- ✓ Accessing items
- ✓ Add, remove, edit key-value pairs
- ✓ Operations on dictionary
- ✓ Dictionary functions
- ✓ Dictionary comprehension

Python Functions

- ✓ Create function
- ✓ Arguments and parameters
- ✓ args and kwargs
- ✓ How functions are executed in a memory
- ✓ Variable scope
- ✓ Deletion of function
- ✓ Returning of function
- ✓ Advantages of functions
- ✓ Lambda functions
- ✓ map(), filter(), reduce()

Object Oriented Programming (OOP)

OOP Part1

- ✓ What is OOP?
- ✓ What are classes and Objects?
- ✓ Banking application coding
- ✓ Methods vs Functions
- ✓ Class diagram
- ✓ What is the true benefit of constructor?
- ✓ Concept of 'self'

OOP Part2

- ✓ Revision of last session by solving problems
- ✓ How objects access attributes
- ✓ Attribute creation from outside of the class
- ✓ Reference Variables
- ✓ Mutability of Object
- ✓ Encapsulation
- ✓ Collection of objects
- ✓ Static variables and methods

OOP Part3

- ✓ Method Overriding
- ✓ Super keyword
- ✓ Super constructor

- ✓ Practice questions on Inheritance
- ✓ Types of Inheritance (Single, Multilevel, Hierarchical, Multiple)
- ✓ Hybrid Inheritance
- ✓ Code example and diamond problem
- ✓ Polymorphism
- ✓ Method Overriding and Method Overloading

Session on Abstraction

- ✓ What is Abstraction?
- ✓ Bank Example Hierarchy
- ✓ Abstract class
- ✓ Coding abstract class (BankApp Class)

Advanced Python:

File Handling + Serialization & Deserialization

- ✓ How File I/O is done
- ✓ Writing to a new text file
- ✓ What is open()?
- ✓ append()
- ✓ Writing many lines
- ✓ Saving a file
- ✓ Reading a file - read() and readline()
- ✓ Using context manager - with()
- ✓ Seek and tell
- ✓ Serialization and Deserialization
- ✓ Pickling

Exception Handling

- ✓ Syntax Error with Examples
- ✓ Exception with Examples
- ✓ Why we need to handle Exception?
- ✓ Exception Handling (Try-Except-Else-Finally)
- ✓ Handling Specific Error
- ✓ Raise Exception
- ✓ Create custom Exception

Decorators and Namespaces

- ✓ Namespaces
- ✓ Scope
- ✓ Hands-on local, enclosing, global and built-in scope
- ✓ Decorators with Examples

Iterators

- ✓ What are iterators
- ✓ What are iterables
- ✓ How for loop works in Python?
- ✓ Practical example to use iterator

Generator

- ✓ What is a generator?
- ✓ Why to use Generator?
- ✓ Yield vs Return
- ✓ Generator Expression

    - ✓ Practical Examples
    - ✓ Benefits of generator

  Logging

  PEP8 Guidelines

PYTEST Library

- ✓ Introduction and importance of testing
- ✓ TDD approach
- ✓ Testcases writing
- ✓ Test code coverage

Database Connectivity:

- ✓ MySQL Introduction and Installation
- ✓ CREATE DB,
- ✓ DROP DB
- ✓ CREATE TABLE,
- ✓ DELETE TABLE
- ✓ SELECT WHERE, NOT, AND, OR ORDER BY
- ✓ INSERT INTO
- ✓ Handling NULL UPDATE, DELETE
- ✓ LIMIT COUNT, MIN, MAX, LIKE
- ✓ IN JOINS GROUP BY
- ✓ ALTER TABLE PRIMARY KEY, FOREIGN KEY

Pandas:

- ✓ Introduction
- ✓ Series and DataFrames
- ✓ read files using pandas
- ✓ find duplicates and null values
- ✓ create a columns
- ✓ apply on dataframe
- ✓ join and merge dataframes
- ✓ saving dataframe to a file.