Python Programming – Basic to advanced

Duration: 5 days

Day 1: Python Core Programming Foundations

Module 1: Introduction to Python and Basic Data Types

- - Python Interpreter & Environment Setup
- - Python 3.x Features & Relevance
- - Variable Declaration & Dynamic Typing
- - Data Types: Numbers, Strings, Booleans
- - String Quotations, Escape Sequences

Module 2: Control Structures

- - Conditional Statements: if, elif, else
- - Looping Constructs: while, for
- - Loop Control: break, continue, pass, else in loops

Module 3: Core Data Structures

- - Lists, Tuples, Arrays
- - Dictionaries & Sets
- - Slicing & Indexing Techniques

Module 4: Functions

- - Defining & Calling Functions
- - Parameters: Positional, Keyword, Default, Arbitrary Args
- - Return Values
- - Lambda Expressions
- - Unpacking Argument Lists
- - Docstrings

Module 5: Functional Programming Tools

- - lambda, map, filter, reduce
- - List Comprehensions
- - Built-in functions: isalpha, apply (apply is obsolete in Python 3)

Day 2: Intermediate Python – Files, OOP, and Exceptions

Module 6: File Handling & OS Interactions

- - File Operations: Open, Read, Write, Append, Close
- - File Modes & Context Manager (with)
- - Working with CSV Files

Module 7: Python Modules

- - Creating & Importing Modules
- - Module Search Path & __name__ == '__main__'

- - Compiled .pyc files
- - Standard Modules Overview: math, random, etc.
- - Using dir() for Introspection

Module 8: Object-Oriented Programming

- - Class & Object Basics
- - The self keyword
- - Constructor (__init__) & Destructor
- - Instance vs Class Variables
- - Inheritance Basics
- - Method Types: Instance, Class, Static

Module 9: Exception Handling

- - Built-in Exceptions
- - try, except, else, finally
- - Raising Exceptions
- - Creating Custom Exceptions

Day 3: Advanced Python Concepts

Module 10: Supplementary Topics

- - Closures
- - Basic Debugging (pdb, print debugging)
- - Pickle for Object Serialization
- - File Compression (gzip, zipfile)
- - Useful Modules: os, sys, pprint

Module 11: Regular Expressions

- - Pattern Matching Basics
- - Special Characters, Quantifiers, Anchors
- - Compilation using re.compile()
- - Search, Match, Substitution
- - Group Extraction

Module 12: MS Excel Automation with Python

- - Installing openpyxl / pandas / xlrd
- - Reading & Writing Excel Files
- - Working with different data structures (List, Dict, DF)

Module 13: REST API Interaction

- - Introduction to JSON format
- - Accessing APIs with urllib3 & requests
- - Making GET and POST Requests
- - Parsing JSON Responses

Module 14: MySQL Database Connectivity

- - Using mysql-connector / PyMySQL
- - Establishing Connection
- - Performing CRUD Operations
- - Error Handling

Day 4: Concurrency and Data Analysis with Pandas & NumPy

Module 15: Multithreading & Multiprocessing

- - Threading vs Multiprocessing
- - Python GIL Overview
- - threading Module: Threads, Daemon Threads
- - multiprocessing Module: Process, Pipe, Queue, Lock
- - Shared State & IPC
- - Process Exceptions & Handling

Module 16: NumPy – Numerical Computing

- - Arrays & Array Creation
- - Indexing, Slicing, Iteration
- - Array Operations
- - Boolean Indexing
- - Shape & Dimensional Manipulations
- - Reading & Writing Arrays to Files

Module 17: Pandas – Data Manipulation & Analysis

- - Data Structures: Series, DataFrames
- - Hierarchical Indexing
- - Filtering, Querying & Slicing
- - Aggregation & Grouping
- - Data Merging & Joining

Day 5: Python Web Development with Django & Testing

Module 18: Django Web Framework

- - Django Overview & Installation
- - Project Structure & MVC Pattern
- - Running Django Development Server
- - Creating First Project

Module 19: Django Models

- - Model Class & Field Definitions
- - Field Types & Options
- - File Handling in Models
- - Signals and Model Events

- - Model Inheritance

Module 20: Views, Templates & Forms

- - Creating Views & HTML Templates
- - Connecting Models with Templates
- - Using Django Forms & CSRF Protection

Module 21: Unit Testing in Python

- - Unit Testing Fundamentals
- - unittest Module
- - Writing Test Cases for Functions & Classes
- - Automating Tests with Test Suites