

## PYTHON FULL-STACK DEVELOPMENT

### Module 1: Core Python

#### Introduction

- History
- Features
- Setting up path
- Working with Python Basic Syntax

#### Data Types Introduction

- Words order, unordered, sequential, non-sequential, mutable and immutable
- List, Tuple, String, Dictionaries, Set, Frozenset – Introduction
- Accessing – List, Tuple, String, Dictionaries, Set, Frozenset
- Shallow and deep copy

#### Conditional Statements

- If, If-else, Nested if-else

#### Looping

- For, While, Nested loops, For-else

#### Control Statements

- Break, Continue, Pass

#### Functions

- Defining, Calling functions
- Function Arguments
- Global & local variables

#### Higher Order Functions

- List, Dict, Generator Expressions / Tuple Comprehension
- Map, Filter, Reduce, Lambda (Anonymous functions)

#### File Operation

- Open and With open
- Map, Filter, Reduce, Lambda (Anonymous functions)

## Input-Output

- Printing on screen
- Reading data from keyboard

## Parsers

- JSON, CSV, XML, Serialization
- 

## Module 2: Advance Python

### OOPs Concepts

- Class
- Object
- Inheritance
- Abstract
- Polymorphism
- Encapsulation

### Linked Concepts

- Types of Methods (Instance, Static, Class)
- MRO
- Operator Overloading
- Descriptors (Property – setter, getter)

### Modules

- Module, Package, Importing modules
- base64
- OS, Sys, Path
- Datetime
- Collections
- Argparse
- Logging
- Email sending
- Pdb and ipdb
- Subprocess
- Async
- Random

## Exception Handling

- Exceptions and Handling
- User Defined Exceptions

## Concepts

- Iterators, Generators, Closures, Decorators

## Multithreading

- Thread, Starting a thread
- Threading module
- Synchronizing threads
- Multithreaded Priority Queue

## Regular Expressions

- Match, Search, findall, split, sub, Patterns

## Database

- Introduction
- Connections
- Executing queries
- Transactions
- Handling errors

---

## Module 3: Django

### Introduction

- What is Django
- Django and Python
- Django's take on MVC
- How to get and install Django

### Getting Started

- models.py, views.py, urls.py

### Migrations

- Introduction to Migrations
- Data Migrations

## Django URL Patterns and Views

- Designing a Good URL Scheme
- Generic Views

## Django Forms

- Form Classes
- Validations
- Authentication

## Django and REST APIs

- Django REST-API
- Django Piston

## Unit Testing with Django

- Overview of Unit Testing
- Using unittest2 Library
- Tests and Test Databases
- Doctests and Debugging Best Practices

---

## Module 4: Linux

- Installing Linux (Dual Boot and Virtual Machines)
- Kernel, Shell basics
- Navigating Linux File System
- CLI basics & common commands (ls, cd, cp, rm, mv)
- File and Directory Operations
- File Permissions and Ownership
- Process Management (ps, top, kill)
- Package Management (apt, yum, repositories)
- Text Editing (Nano, Vi)
- Introduction to Shell Scripting
- Networking (ping, ifconfig, netstat, SSH, firewalls)
- User and Group Management
- Disk and Storage Management
- System Monitoring and Logs
- Linux Security (passwords, SELinux, AppArmor)
- Real-world exercises (Web server setup, automation scripts, troubleshooting)
- Course Review & Advanced Learning Resources

---

## Module 5: Selenium

- Introduction to Automation & Testing
- What is Selenium?
- WebDriver Overview & Components
- Configuring WebDriver with different browsers
- WebDriver Manager
- Locators – Normal, Custom, CSS Selectors, Xpaths, Functions, Axes
- Forms – Textboxes, Radio buttons, Checkboxes, File upload
- Select class – Dropdowns
- Action Chains – Drag & Drop, Keyboard, Mouse Actions
- Tables – Data picker
- Browser Options – Headless, Popups, Developer options
- Pagination
- Window Handling – Alert popups
- Pytest
- Page Object Model

---

## Module 6: GIT – Version Control

- Version Control Overview
- Git History and Purpose
- Git vs Other Systems
- Installing Git (Windows, Linux, macOS)
- Configuration – User Name & Email
- Git Interfaces (CLI, GUI)
- Repositories and Three States
- Creating First Repository
- Git Commands – init, add, commit, status, log, diff
- Branching & Merging
- Merge Conflicts
- Working with Remote Repositories
- Cloning, Pushing, Pulling
- Forks and Pull Requests
- Collaborative Workflows
- Advanced Git – Stashing, Tagging, Rebasing
- Undoing Changes – Revert, Reset, Clean
- Git Hosting Services – GitHub, GitLab, Bitbucket
- Remote Repo Management & CI/CD Basics

- Best Practices – Commit Messages, Clean History
  - Branch Management Strategies
  - Practical Exercises & Projects
  - Open Source Contribution
  - Advanced Git Topics (Submodules, Hooks)
- 

## Module 7: Database

- Why Database? Importance in Computing
  - Data Types Overview
  - Relational Database Management System (RDBMS)
  - Relational Algebra and Calculus
  - Entity-Relationship Model
  - Normalization and Denormalization
  - SQL Basics – CREATE, SELECT, INSERT, UPDATE, DELETE
  - Advanced SQL – GROUP BY, HAVING, JOINS
  - Operators – IN, LIKE, ISNULL, UNION
  - Views, Indexes, Triggers
  - Subqueries (Inner & Outer)
  - Data Modeling with SQL
  - Analytical Functions – Rank, Dense Rank, Partition By
  - Database Design – Process, Dependencies, Normalization
  - Table Design and Relationships
  - NoSQL Databases – Document, Key-Value, Graph, Column-Family
  - RDBMS vs NoSQL comparison
  - Database Management & Transactions
  - ACID Properties
  - Concurrency Control and Isolation Levels
- 

## Module 8: Soft Skills

- Communication Skills – Verbal, Written, Presentations
  - Teamwork & Collaboration
  - Time Management – Prioritization, Deadlines
  - Problem-Solving & Critical Thinking
  - Adaptability & Flexibility
  - Professional Etiquette – Workplace behavior, Email writing, Meetings
  - Interview Preparation – Mock interviews, Resume building, Self-introduction
-

**\*\*\*Note:** This outline is comprehensive and can be tailored based on course duration, depth of coverage, and the participants expertise levels. As technology continues to evolve, it is crucial to review and update the content regularly to incorporate emerging tools, practices, and industry best standards.

