

# Python Programming

1

## Module 1: Introduction to Python

### 1. Overview of Python

- What is Python?
- History and Evolution of Python
- Features of Python
- Applications of Python

### 2. Setting Up the Environment

- Installing Python
- Downloading Python from python.org
- Installation on Windows, macOS, and Linux
- Installing and Using an IDE (e.g., PyCharm, VSCode)
- Introduction to the Python Interpreter and Interactive Mode
- Writing and Running Your First Python Script

2

## Module 2: Basic Python Syntax

### 1. Python Basics

- Python Syntax and Indentation
- Comments in Python
- Print Statement and String Formatting

Pg. 1



# Python Programming

## 2. Variables and Data Types

- Variables and Assignments
- Basic Data Types (int, float, str, bool)
- Type Conversion and Casting
- Basic Input and Output

## 3. Basic Operators

- Arithmetic Operators (+, -, \*, /, %, \*, //)
- Comparison Operators (==, !=, >, <, >=, <=)
- Logical Operators (and, or, not)
- Assignment Operators (+=, -=, \*=, /=)

3

## Module 3: Control Flow

### 1. Conditional Statements

- if, elif, and else Statements
- Nested Conditions
- Ternary Conditional Operator

### 2. Loops

- For Loops
- While Loops
- Break and Continue Statements
- Nested Loops

Pg. 2

# Python Programming

## 3. Comprehensions

- List Comprehensions
- Dictionary Comprehensions

4

## Module 4: Functions and Modules

### 1. Functions

- Defining and Calling Functions
- Function Parameters and Arguments
- Return Values
- Variable Scope (Local and Global)
- Lambda Functions

### 2. Modules and Packages

- Importing Modules
- Standard Library Modules (e.g., `math`, `datetime`)
- Creating and Using Custom Modules
- Understanding Packages and `\_\_init\_\_.py`

Pg. 3



# Python Programming

5

## Module 5: Data Structures

### 1. Lists

- Creating and Accessing Lists
- List Operations and Methods
- Slicing and Indexing

### 2. Tuples

- Creating and Accessing Tuples
- Tuple Operations and Methods

### 3. Dictionaries

- Creating and Accessing Dictionaries
- Dictionary Operations and Methods
- Iterating through Dictionaries

### 4. Sets

- Creating and Accessing Sets
- Set Operations and Methods

Pg. 4

# Python Programming

6

## Module 6: File Handling

### 1. Reading and Writing Files

- Opening and Closing Files
- Reading from Files
- Writing to Files
- File Modes (r, w, a, r+)

### 2. Working with File Paths

- Understanding File Paths
- Using `os` and `pathlib` for File Operations

7

## Module 7: Error Handling and Exceptions

### 1. Exception Handling

- What are Exceptions?
- try, except, else, and finally Blocks
- Raising Exceptions

### 2. Common Python Exceptions

- Handling Specific Exceptions (e.g., ValueError, IndexError)

Pg. 5



# Python Programming

8

## Module 8: Introduction to Object-Oriented Programming (OOP)

### 1. Basic Concepts of OOP

- Classes and Objects
- Attributes and Methods

### 2. Creating and Using Classes

- Defining a Class
- Initializing Objects with `\_\_init\_\_` Method
- Instance vs. Class Methods
- Inheritance and Polymorphism

### 3. Encapsulation and Abstraction

- Access Modifiers (Public, Protected, Private)
- Abstract Classes and Methods

Pg. 6



# Python Programming

9

## Module 9: Advanced Topics

### 1. Database Connection with MySQL

- Introduction to MySQL
- Installing MySQL and MySQL Connector for Python
- Connecting to a MySQL Database
- Executing SQL Queries
- Fetching and Inserting Data
- Handling Database Transactions

### 2. Multithreading

- Understanding Multithreading
- Creating and Using Threads
- Synchronization and Thread Safety
- Practical Examples

### 3. Iterators and Generators

- Introduction to Iterators
- Creating and Using Iterators
- Understanding Generators
- Creating and Using Generators

Pg. 7



# Python Programming

## 4. Decorators

- Introduction to Decorators
- Function Decorators
- Class Decorators
- Practical Examples

## 5. Regular Expressions (Regex)

- Introduction to Regex
- Basic Regex Patterns
- Using `re` Module for Pattern Matching
- Practical Examples

## 6. JSON and XML Processing

- Introduction to JSON and XML
- Parsing JSON Data
- Parsing XML Data
- Working with `json` and `xml.etree.ElementTree` Modules

## 7. Overview of Advanced Concepts

- Brief Overview of Advanced Python Topics (e.g., Async Programming, Memory Management)
- Resources for Further Learning

Pg. 8



# Python Programming

10

## Module 10: Final Project and Review

### 1. Capstone Project

- **Project Overview:** Develop a **Personal Finance Manager** application.
- **Project Requirements:**
  - Users can add, edit, and delete financial transactions.
  - The application should categorize transactions and display a summary report.
  - Store data in a MySQL database.
  - Implement a multithreaded feature for background data processing.
  - Use decorators for validation of user input.
  - Implement basic error handling and data validation.
- **Project Goals:**
  - Apply concepts learned in the course.
  - Integrate multiple components such as file handling, database operations, and GUI (optional).
  - Demonstrate the ability to build a complete application.

Pg. 9



# Python Programming

10

## Module 10: Final Project and Review

### 2. Review and Q&A

- Reviewing Key Concepts
- Addressing Questions and Clarifications
- Discussing the Capstone Project and Future Learning Paths

---

This syllabus provides a thorough foundation in Python programming, from basic to advanced topics, and includes a capstone project with a MySQL database integration to showcase practical application of the learned skills.

Pg. 10

