Python Course Modules

Course Overview

This course introduces learners to Python, one of the most versatile and widely used programming languages today. Designed for beginners as well as those looking to solidify their programming foundation, the course covers the core principles of Python programming and its practical applications in various domains. Through hands-on exercises and real-world projects, students and professionals will gain practical experience in writing efficient Python code for tasks such as automation, data processing, and simple application development.

Learning Objectives

- Understand the basic syntax and structure of Python programs.
- Write Python code using variables, data types, and operators.
- Work with built-in data structures: lists, tuples, dictionaries, and sets.
- Handle errors and exceptions using try and except blocks.
- Perform basic file input/output (read/write files).
- Write simple programs that solve real-world problems.
- Use Python IDLE, Jupyter Notebook, or other IDEs for coding and testing.

Prerequisites

1. Basic proficiency in mathematics and familiarity with spreadsheets.

2. No prior programming experience required.

Course Modules

Modules	Title	Topics
1.	**Introduction to Python**	➢ History and features of Python ➢ Installing Python and setting up the environment ➢ Python IDEs (IDLE, Jupyter, VS Code) ➢ Writing and executing first Python program
2.	**Python Basics**	➢ Variables and data types ➢ Input/output functions ➢ Type casting and operators ➢ Comments and code readability
3.	**Control Flow**	➢ Conditional statements (if, elif, else) ➢ Looping structures (for, while)

		➢ Loop control statements (break, continue, pass)
4.	**Data Structures**	➢ Strings and string methods ➢ Lists and list operations ➢ Tuples, sets, and dictionaries ➢ Nested structures and comprehensions
5.	**Functions and Modules**	➢ Defining and calling functions ➢ Arguments and return values ➢ Lambda functions ➢ Importing and using modules ➢ Creating custom modules
6.	**File Handling**	➢ Reading and writing text files ➢ Working with CSV files ➢ File modes and file operations ➢ Exception handling with file I/O
7.	**Object-Oriented Programming (OOP)**	➢ Classes and objects ➢ Constructors and destructors ➢ Inheritance and polymorphism ➢ Encapsulation and abstraction
8.	**Error Handling**	➢ Types of exceptions ➢ try, except, finally, and raise ➢ Creating custom exceptions
9.	**Working with Libraries**	➢ Introduction to NumPy for numerical computing ➢ Using Pandas for data manipulation ➢ Matplotlib for data visualization
10.	**Advanced Python Concepts (Optional/Bonus)**	➢ List comprehensions and generators ➢ Decorators and closures ➢ Working with APIs and JSON ➢ Basics of multithreading and multiprocessing
11.	**Project Work**	➢ Real-world mini projects ➢ Code review and presentation ➢ Final assessment