Frimity_Edutech

Grow With Us

Course: Python Programming

Course Description:

This course provides a comprehensive introduction to the Python programming language, focusing on its versatility and ease of use. Students will learn the fundamentals of Python syntax, data structures, and programming concepts through hands-on exercises and projects.

Course Objectives:

- Understand the core principles and features of the Python programming language.
- Gain proficiency in Python syntax, data types, and control structures.
- Develop skills to design and implement Python programs.
- Learn how to work with Python libraries and Weeks.
- Explore best practices and coding standards for Python programming.

Course Outline:

Week 1: Introduction to Python along with Python Syntax and Language Fundamentals

- Overview of Python and its role in software development.
- Setting up the Python development environment.
- Introduction to Python development tools (e.g., PyCharm, Jupyter Notebook).
- Variables, data types, operators, and expressions.
- Control structures: if-else, loops, and conditional statements.
- Input/output and exception handling.

Week 2: Functions and Weeks as well as Data Structures in Python

- Defining and using functions in Python.
- Function parameters and return values.
- Organizing code into Weeks and packages.
- Lists, tuples, sets, and dictionaries.
- Manipulating and iterating over data structures.
- List comprehensions and generator expressions.

Week 3: Object-Oriented Programming in Python along with File Handling and Data Serialization

- Principles of object-oriented programming (OOP) in Python.
- Classes, objects, and methods.
- Inheritance, encapsulation, and polymorphism.
- Reading from and writing to files in Python.

- Working with different file formats (e.g., CSV, JSON).
- Serialization and deserialization of data.

Week 4: Python Libraries and Packages as well as Error Handling and Debugging

- Exploring commonly used Python libraries and packages.
- Working with libraries for data manipulation, scientific computing, or web development.
- Installation and management of third-party packages using pip.
- Understanding exceptions and exception handling in Python.
- Techniques for effective error handling and debugging.
- Logging and error reporting.

Week 5: Python Standard Library along with Web Scraping and Automation

- Exploring the Python Standard Library.
- Working with Weeks for string manipulation, regular expressions, date/time, and more.
- Leveraging built-in functionality for efficient programming.
- Introduction to web scraping using Python.
- Parsing HTML and extracting data from websites.
- Automating tasks using scripting and scheduling.

Week 6: Data Analysis and Visualization

- Introduction to data analysis using Python.
- Working with libraries like NumPy and Pandas for data manipulation.
- Data visualization using Matplotlib or other plotting libraries.

Week 7: Project Development

- Applying learned concepts and techniques to develop a Python project.
- Collaborative project development in teams.
- Project management and software development lifecycle.

Grading Criteria:

- Quizzes and Assignments: 40%
- Python Programming Projects: 50%
- Class Participation: 10%