55264-A: Introduction to Programming Using Python

Course Duration: 40 Hours (5 Days)

Overview

The 55264-A: Introduction to Programming Using Python course is a comprehensive paid python course that provides learners with the foundational skills necessary to start programming in Python. Designed for beginners, this intro to python programming course covers essential concepts such as Data types, Operators, Control flow, Input/output operations, Code documentation, Error handling, and the Use of modules. Through a series of modules, each consisting of lessons and hands-on labs, participants will learn to write Python code effectively and solve real-world problems. The structured learning ensures that by the end of the course, learners are proficient in Writing Python code, performing operations, Debugging, and leveraging Python's extensive ecosystem of modules and tools, thus setting a solid foundation for further programming endeavors and career advancement.

Audience Profile

Course 55264-A: Introduction to Programming Using Python is designed for beginners looking to build foundational skills in Python programming.

- Aspiring programmers and developers
- Data analysts and data scientists
- Students pursuing computer science or software engineering
- Technical professionals seeking to upskill
- Hobbyists interested in learning programming
- IT professionals needing to automate tasks
- Business analysts aiming to understand data manipulation
- Educators and teachers preparing to teach programming
- Quality assurance testers looking to write automation scripts
- System administrators interested in scripting for task automation
- Non-technical professionals seeking a basic understanding of programming concepts

Course Syllabus

Module 1: Performing Operations Using Data Types and Operators

- This module explains how to use Python operators and data types to achieve specified results.

Lessons:

- Assigning data types to variables
- Performing data and data type operations

- Executing arithmetic, comparison, and logical operations
- Review

Lab: Performing Operations Using Data Types and Operators

- Assigning data types to variables
- Performing data and data type operations
- Executing arithmetic, comparison, and logical operations

After completing this module, students will be able to:

- Assign data types to variables
- Manage data and perform operations on different data types
- Execute arithmetic, comparison, and logical operations

Module 2: Control Flow with Decisions and Loops

- This module explains how to use control flow and looping operations in Python.

Lessons:

- Constructing and analyzing code segments that use branching statements
- Constructing and analyzing code segments that perform iterations
- Review

Lab: Control Flow with Decisions and Loops

- Implementing branching operations
- Implementing iteration operations

After completing this module, students will be able to:

- Implement branching operations
- Implement iteration operations

Module 3: Performing Input and Output Operations

- This module explains how to construct input and output operations using files and the console.

Lessons:

- Writing Python code segments to perform file input and output operations
- Writing Python code segments to perform console input and output operations
- Review

Lab: Performing Input and Output Operations

- Performing input and output operations using files

- Performing input and output operations from the console

After completing this module, students will be able to:

- Perform input and output operations using files
- Perform input and output operations via the console

Module 4: Documenting and Structuring Code

- This module explains how to structure and document well-written Python code.

Lessons:

- Writing and analyzing structured code segments
- Documenting code segments using comments and documentation strings
- Review

Lab: Documenting and Structuring Code

- Writing and analyzing code segments
- Documenting code segments

After completing this module, students will be able to:

- Write structured code
- Document code effectively using comments and documentation strings

Module 5: Troubleshooting and Error Handling

- This module explains how to troubleshoot and handle errors in Python.

Lessons:

- Analyzing, detecting, and fixing code errors
- Writing code to handle exceptions effectively
- Review

Lab: Troubleshooting and Error Handling

- Analyzing, detecting, and fixing code errors
- Writing and implementing exception-handling code

After completing this module, students will be able to:

- Identify and fix errors in Python code
- Implement error-handling mechanisms

Module 6: Performing Operations Using Modules and Tools

- This module explains how to use built-in Python modules for various operations.

Lessons:

- Using built-in modules for basic operations
- Using built-in modules for complex operations
- Review

Lab: Performing Operations Using Modules and Tools

- Utilizing built-in modules for basic operations
- Utilizing built-in modules for complex operations

After completing this module, students will be able to:

- Use built-in modules to perform operating system, date, and mathematical operations