15-Day Python Programming Course

Day 1-2: Introduction to Python and Python Basics

- Overview of Python, Features, and Applications
- Setting up the environment (Installing Python and IDEs)
- Writing and running the first Python script
- Python Syntax and Code Structure
- Variables, Data Types, Constants
- Input and Output

Day 3-4: Operators and Control Flow Statements

- Operators in Python (Arithmetic, Relational, Logical, Bitwise)
- Conditionals (if, elif, else)
- Looping Constructs (for, while loops)
- Loop Control (break, continue, pass)

Day 5-6: Functions and Data Structures

- Functions in Python (arguments, return values, scope)
- Data Structures (Lists, Tuples, Dictionaries, Sets)
- List, Dictionary, and Set Comprehensions

Day 7-8: File Handling

- Reading and Writing Text Files
- File modes and operations (open, with)
- Working with Binary and CSV Files

Day 9-10: Modules, Libraries, and Error Handling

- Importing and using built-in modules (math, random, etc.)
- Installing and using third-party libraries (NumPy, pandas)
- Custom modules
- Error handling with try-except

Day 11-15: Project & Review

- Implementing a project using learned concepts
- Code debugging and optimization
- Final Q&A session and review

30-Day Python Programming Course

Week 1: Introduction to Python and Python Basics

- Overview of Python, Features, and Applications
- Setting up the environment and writing the first Python script
- Variables, Data Types, and Constants
- Operators (Arithmetic, Relational, Logical)

Week 2: Control Flow Statements and Functions

- Conditional Statements (if, elif, else)
- Looping Constructs (for, while loops)
- Functions in Python (arguments, return values, scope)

Week 3: Data Structures

- Lists, Tuples, Dictionaries, and Sets
- Comprehensions (List, Dictionary, Set)
- File Handling: Text, Binary, CSV files

Week 4: Modules, Libraries, and Error Handling

- Importing and using built-in modules (math, os, etc.)
- Installing third-party libraries (NumPy, pandas)
- Custom modules and Error handling (try, except, raise)

Day 29-30: Project & Review

- Working on a project using the learned concepts
- Code review and final Q&A session

45-Day Python Programming Course

Week 1-2: Introduction to Python and Python Basics

- Overview of Python, Features, Applications, and Setup
- Variables, Data Types, Constants, and Operators

Week 3: Control Flow Statements

- Conditional Statements
- Looping Constructs (for, while)
- Loop Control Statements (break, continue)

Week 4: Functions and Data Structures

- Functions (Arguments, Return Values, Scope, Lambda Functions)

- Data Structures: Lists, Tuples, Dictionaries, Sets

Week 5: File Handling and Modules

- File Operations (Text, Binary, CSV Files)
- Working with Python Libraries (math, random, os)
- Third-Party Libraries (NumPy, pandas)

Week 6: Advanced Concepts & Error Handling

- Custom modules and packages
- Error and Exception Handling

Day 41-45: Project and Final Review

- Develop a project utilizing Python concepts
- Debugging and code optimization
- Final review and Q&A

60-Day Python Programming Course

Week 1-2: Introduction and Basics

- Introduction to Python and Setup
- Python syntax, variables, and data types
- Operators and basic syntax rules

Week 3-4: Control Flow, Functions, and Data Structures

- Conditional statements and loops
- Functions (arguments, return values, scope)
- Data Structures (Lists, Tuples, Dictionaries, Sets)

Week 5: File Handling and Working with Libraries

- Reading and writing text, binary, and CSV files
- Modules and Libraries (math, random, os, pandas, NumPy)

Week 6-7: Advanced Topics in Functions and Error Handling

- Lambda functions, error handling (try, except)
- Creating custom modules

Week 8: Project Development

- Work on an independent project

- Applying learned concepts

Day 57-60: Final Review and Assessment

- Review of concepts and code optimization
- Final project presentation and Q&A