Phone: +91 8882618533
Email: info@geekinstitute.org
Website: www.geekinstitute.org

THE GEEK INSTITUTE OF CYBER SECURITY

Regd. By: E-Max India
Centre Code: EMAX/EK80606
(Recognized By Govt. Of India)

(Building Futures Through Digital Knowledge and Innovation)

Python Programming (03 Months) Syllabus

01: Introduction to Python

- ✓ What is Python? Benefits and applications
- ✓ Python 2 vs Python 3
- ✓ Installing Python & setting up IDE (IDLE, VS Code, PyCharm)
- ✓ Writing your first Python script
- ✓ Understanding Python syntax, comments, and indentation

02: Python Basics

- ✓ Variables, constants, and data types
- ✓ Type casting and dynamic typing
- ✓ Input/output handling (input(), print())
- ✓ Operators: arithmetic, comparison, logical, bitwise, identity, and membership
- ✓ Operator precedence and expressions

03: Control Flow and Looping

- ✓ if, if-else, elif statements
- ✓ for and while loops with range
- ✓ Loop control: break, continue, pass
- ✓ Nested conditions and loops
- ✓ Flowchart logic and dry runs

04: Data Structures in Python

- ✓ Strings and common string methods
- ✓ Lists and list slicing/indexing
- ✓ Tuples and immutability
- ✓ Sets and set operations
- ✓ Dictionaries and key-value operations
- ✓ Comprehensions: list, dict, set with conditions

5: Functions and Recursion

- ✓ Defining and calling functions
- ✓ Parameters and arguments: default, keyword, *args, **kwargs
- ✓ Return statements and multiple return values
- ✓ Variable scopes: local, global, and nonlocal
- ✓ Lambda functions
- ✓ Recursion with practical examples

06: Modules and Standard Library

- ✓ Importing built-in modules: math, random, datetime, os, sys, etc.
- ✓ Creating and using custom modules
- ✓ Organizing reusable code using packages
- ✓ __main__ and script/module behavior

07: Exception Handling

- ✓ Types of errors: syntax vs runtime
- ✓ Handling exceptions using try, except, else, finally
- ✓ Raising exceptions manually
- ✓ Creating custom exceptions
- ✓ Real-world use cases of exception handling

08: File Handling

- ✓ Opening, reading, writing files
- ✓ File modes and cursor management
- ✓ Working with binary files
- ✓ with statement and file contexts
- ✓ Working with CSV files using csv module
- ✓ Working with JSON files using json module

09: Object-Oriented Programming (OOP)

- ✓ Classes and objects
- ✓ Constructors (__init__) and destructors
- ✓ Instance and class variables
- ✓ Methods: instance, static, class
- ✓ Inheritance (single, multiple, hierarchical)
- ✓ Polymorphism and method overriding
- ✓ Abstraction, encapsulation
- ✓ Magic methods (__str__, __repr__, etc.)

10: Advanced Python Concepts

- ✓ List, dict, and set comprehensions with conditions
- ✓ Generators and yield keyword
- ✓ Iterators and __iter__, __next__
- ✓ Decorators
- ✓ Closures and higher-order functions
- ✓ Functional programming: map(), filter(), reduce()

11: GUI Programming with Tkinter

- ✓ Introduction to GUI development
- ✓ Creating GUI windows
- ✓ Adding widgets: Label, Entry, Button, TextBox
- ✓ Layout management (pack, grid)
- ✓ Event-driven programming
- ✓ Simple GUI project (e.g., Calculator / Login Form)

12: Python with Databases

- ✓ Using sqlite3 and MySQL with Python
- ✓ Connecting to databases
- ✓ Executing SQL queries from Python
- ✓ Creating and managing tables
- ✓ Performing CRUD operations
- ✓ Database project: Student Management System

13: Web Scraping Basics

- ✓ Understanding HTTP, URLs, HTML DOM
- ✓ Using requests to fetch web data
- ✓ Parsing HTML with BeautifulSoup
- ✓ Navigating elements, extracting tags and data
- ✓ Saving data to CSV/Excel
- ✓ Project: Web scraper for job listings or product data

14: Consuming APIs and JSON

- ✓ REST API basics
- ✓ Making HTTP requests with requests
- ✓ Parsing and handling JSON data
- ✓ Consuming third-party APIs (e.g., weather, currency)
- ✓ API integration mini-project

15: Automation and Scripting

- ✓ Automating tasks like file renaming, sorting, backups
- ✓ Sending emails with SMTP and smtplib
- ✓ Working with Excel using openpyxl
- ✓ Scheduling jobs with schedule and time
- ✓ Real-world automation project

Module 16: Testing and Debugging

- ✓ Debugging with print() and logging
- ✓ Using pdb debugger
- ✓ Writing test cases with unittest
- ✓ Assertions and test-driven development basics

17: Git and GitHub Basics

- ✓ Version control with Git
- ✓ Git commands: init, add, commit, push, pull, branch, merge
- ✓ Hosting projects on GitHub
- ✓ Managing repositories and collaboration

18: Data Structures and Algorithms (DSA) with Python

- ✓ Importance of DSA for interviews and problem solving
- ✓ Big O notation: time and space complexity
- ✓ Arrays and list-based manipulations
- ✓ Searching algorithms: linear, binary
- ✓ Sorting: bubble, selection, insertion, merge, quick
- ✓ Stacks and queues using lists and collections.deque
- ✓ Linked lists (single), implementation and traversal
- ✓ Hashing: dictionaries and sets
- ✓ Recursion and backtracking basics
- ✓ Practice problems using LeetCode-style questions

19: Data Analysis with NumPy, Pandas, and Matplotlib

NumPy

- ✓ Installing NumPy and basic usage
- ✓ Arrays, vectorization, slicing, broadcasting
- ✓ Array manipulation, reshaping, stacking
- ✓ Mathematical operations and aggregations

Pandas

- ✓ Installing and importing Pandas
- ✓ Series and DataFrames
- ✓ Reading/writing CSV, Excel, JSON files
- ✓ Data filtering, sorting, and indexing
- ✓ Handling missing data and data cleaning
- ✓ Grouping and aggregation

Matplotlib

- ✓ Plotting basics: line, bar, scatter, histogram
- ✓ Customizing plots: labels, legends, grids, colors
- ✓ Plotting from Pandas DataFrames
- ✓ Saving and exporting plots
- ✓ Mini-project: Data visualization dashboard

20: Final Projects and Interview Preparation

- ✓ **Capstone Project** (choose one or more):
  - Portfolio Website Backend
  - Data Analysis Dashboard
  - Automation Script
  - GUI-based Management System
  - API Consumer Application
- ✓ Python interview questions (basic to advanced)

Course Requirements

- ✓ Basic computer knowledge
- ✓ No prior programming experience required
- ✓ Recommended IDE: VS Code / PyCharm