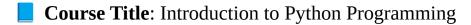
Sangy Academy

www.sangyacademy.com





Duration: 3 Months

Frequency: 2 sessions/week **Session Length**: 2 hours

Total Sessions: 24 **Total Hours**: 48 hours

© Target Audience:

- Beginners in programming
- Students interested in data science, automation, or web development
- · Professionals seeking to automate tasks or learn a versatile language

Sourse Objectives:

By the end of the course, learners will be able to:

- · Understand and use Python syntax and core programming concepts
- · Write clean, logical, and efficient code
- Work with files, modules, and packages
- · Build basic projects and scripts to solve real-world problems

Course Outline

Month 1: Python Basics and Core Concepts

✓ Week 1: Getting Started with Python

Session 1

- What is Python?
- Installing Python and IDEs (IDLE, VS Code, Jupyter)
- Writing your first Python program
- Python syntax and indentation

Session 2

Variables and data types

- Numbers, strings, booleans
- Type conversion and input/output
- · Comments and basic debugging

☑ Week 2: Operators and Control Flow

Session 3

- Arithmetic, comparison, and logical operators
- String operations

Session 4

- if, elif, else statements
- · Nested conditions
- Practical exercises

Week 3: Loops and Iterations

Session 5

- for and while loops
- Loop control: break, continue, pass

Session 6

- · Nested loops
- Pattern and number printing exercises

✓ Week 4: Working with Collections

Session 7

- · Lists: creation, indexing, slicing
- · List methods

Session 8

- Tuples, sets, and dictionaries
- Use cases and operations

Month 2: Functions, Files, and Error Handling

✓ Week 5: Functions in Python

Session 9

- Defining functions
- Parameters and return values
- Scope and global keyword

Session 10

- Default parameters, *args, **kwargs
- · Lambda functions and basic use cases
- Writing utility functions

Week 6: Working with Strings and Files

Session 11

- String formatting and manipulation
- .split(), .join(), .replace(), slicing

Session 12

- File handling: open(), read(), write()
- Reading and writing text files

Week 7: Error Handling and Exceptions

Session 13

- · Types of errors
- Using try, except, finally

Session 14

- Raising exceptions
- Real-world examples: validating input

☑ Week 8: Modules and Packages

Session 15

- Importing built-in modules (math, random, datetime)
- Writing your own modules

Session 16

- Introduction to pip and installing external packages
- Simple script using an external library (e.g. requests)

Month 3: Practical Python and Mini Projects

✓ Week 9: Object-Oriented Programming (OOP) Basics

Session 17

- · Introduction to classes and objects
- __init___, attributes, methods

Session 18

- Inheritance and encapsulation
- Real-life modeling examples

Week 10: Working with Data

Session 19

- · Intro to JSON
- · Working with dictionaries and JSON files
- Data parsing exercises

Session 20

- Introduction to CSV files
- Reading/writing CSVs using CSV module and pandas (optional)

Week 11: Final Project – Part 1

Session 21

- Choose a mini project (e.g., contact book, quiz app, file organizer, or calculator)
- Plan the features and requirements

Session 22

- Start coding with guidance
- Focus on breaking the project into functions and modules

✓ Week 12: Final Project – Part 2 & Presentation

Session 23

- Complete the project
- Add comments and improve code structure

Session 24

- Project presentations
- Feedback and next steps (suggested advanced courses: data analysis, web development, etc.)

Particular Solution Properties **Properties Optional Tools and Libraries**

- Jupyter Notebooks (for interactivity)
- Visual Studio Code or Thonny
- pandas, matplotlib for early exposure to data handling (optional)