

Python Course

Learn Python from Zero to Problem Solving

Welcome to the Python Course!

This course is designed to help you learn Python step by step, starting from the basics and gradually building your programming and problem-solving skills.

You will learn through clear explanations, practical examples, exercises, quizzes, and challenging problems designed to help you understand Python and apply what you learn.

Beginner Level · Self-Paced · Practical Learning

What You Will Learn

By the end of this course, you will be able to:

- Understand the fundamentals of Python programming.
 - Write and run basic Python programs.
 - Work with variables and different data types.
 - Use operators and expressions.
 - Make decisions using conditional statements.
 - Use loops to solve repetitive tasks.
 - Work with strings and Python data structures.
 - Create and use functions.
 - Handle errors and exceptions.
 - Work with files.
 - Understand the fundamentals of Object-Oriented Programming.
 - Solve programming problems using Python.
 - Build small practical projects.
-

Course Roadmap

The course is divided into modules. Each module contains lessons, examples, exercises, and practice questions.

Module 1 — Python Fundamentals

Start your Python journey by learning the basic concepts and syntax of the language.

Topics:

- Introduction to Python
- Python Syntax
- Comments
- Variables
- Variable Naming Rules
- Data Types
- Type Casting
- User Input
- Output
- Basic Python Programs

Practice: Basic Python Exercises

Module 2 — Operators & Expressions

Learn how Python performs calculations, comparisons, and logical operations.

Topics:

- Arithmetic Operators
- Assignment Operators
- Comparison Operators
- Logical Operators
- Identity Operators
- Membership Operators
- Bitwise Operators
- Operator Precedence
- Expressions

Practice: Operators & Expressions Exercises

Module 3 — Conditional Statements

Learn how to make decisions in your programs.

Topics:

- `if` Statements
- `elif` Statements
- `else` Statements
- Nested Conditions
- Multiple Conditions

- Conditional Expressions

Practice: Conditional Statements Challenges

Module 4 — Loops

Learn how to repeat instructions efficiently and control the flow of your programs.

Topics:

- `while` Loops
- `for` Loops
- `range()`
- Nested Loops
- `break`
- `continue`
- `pass`

Practice: Loop Exercises & Challenges

Module 5 — Strings

Learn how to work with and manipulate text in Python.

Topics:

- Creating Strings
- String Indexing
- String Slicing
- String Methods
- String Formatting
- f-Strings
- Escape Characters

Practice: String Exercises

Module 6 — Python Data Structures

Learn how to store, organize, and manipulate collections of data.

Lists

- Creating Lists

- Accessing List Items
- List Indexing
- List Slicing
- List Methods
- Modifying Lists

Tuples

- Creating Tuples
- Accessing Tuple Items
- Tuple Methods
- Tuple Unpacking

Sets

- Creating Sets
- Set Operations
- Set Methods
- Working with Unique Values

Dictionaries

- Creating Dictionaries
- Keys and Values
- Accessing Dictionary Items
- Dictionary Methods
- Updating Dictionaries
- Nested Dictionaries

Practice: Data Structures Exercises

Module 7 — Functions

Learn how to organize your code into reusable blocks.

Topics:

- Defining Functions
- Calling Functions
- Parameters
- Arguments
- Default Arguments
- Keyword Arguments
- Return Values
- Variable Scope
- `*args`
- `**kwargs`

- Lambda Functions

Practice: Functions Challenges

Module 8 — Modules & Packages

Learn how to organize and reuse Python code.

Topics:

- What Is a Module?
- Importing Modules
- Using Built-in Modules
- Creating Your Own Modules
- Introduction to Packages

Practice: Modules Exercises

Module 9 — Error & Exception Handling

Learn how to understand and handle errors in Python programs.

Topics:

- Understanding Errors
- Syntax Errors
- Runtime Errors
- try
- except
- else
- finally
- Raising Exceptions

Practice: Exception Handling Exercises

Module 10 — File Handling

Learn how Python can read from and write to files.

Topics:

- Opening Files
- Reading Files

- Writing to Files
- Appending Data
- File Modes
- Working with Text Files

Practice: File Handling Exercises

Module 11 — Object-Oriented Programming

Build a foundation in Object-Oriented Programming using Python.

Topics:

- Classes
- Objects
- Attributes
- Methods
- The `__init__` Method
- Inheritance
- Encapsulation
- Polymorphism

Practice: Object-Oriented Programming Exercises

Module 12 — Python Problem Solving

Learning syntax is only the beginning.

In this module, you will use everything you have learned to solve programming problems and improve your problem-solving skills.

Challenges include:

- Basic Python Problems
- Conditional Problems
- Loop Challenges
- String Challenges
- List Challenges
- Dictionary Challenges
- Function Challenges
- Mixed Problems
- Advanced Challenges

Final Python Project

Put your knowledge into practice by building a complete Python project.

Learn → Practice → Challenge

This course follows a simple learning approach:

Learn

Understand each concept through clear explanations and practical examples.

Practice

Apply what you learned through exercises and small programming tasks.

Challenge

Test your understanding with increasingly difficult problems.

The goal is not only to memorize Python syntax, but to understand how to use Python to solve problems.

What Is Included?

Throughout the course, you will have access to:

- Step-by-step Python lessons
 - Practical examples
 - Downloadable lecture notes
 - Practice exercises
 - Programming challenges
 - Quizzes
 - Problem-solving activities
 - Solutions and explanations
 - Mini projects
 - A final Python project
-



Who Is This Course For?

This course is suitable for:

- Complete beginners who have never programmed before.
 - Students who want to learn Python from the beginning.
 - Students who want additional practice alongside their university courses.
 - Learners who want to improve their programming and problem-solving skills.
 - Anyone who wants to build a strong foundation in Python.
-



Requirements

No previous programming experience is required.

You will need:

- A computer or laptop
 - Python installed on your device
 - An internet connection
 - A willingness to practice and solve problems
-



Start Your Python Journey

Learning programming takes practice.

Don't worry if the concepts seem difficult at first. Work through each lesson, try the examples yourself, solve the exercises, and don't be afraid to make mistakes.

Every problem you solve is a step toward becoming a better programmer.

Ready to Start?

Start learning Python and build your programming skills step by step.

[ **Start Learning Python**]