Faysal Aziz

# Introduction to Python

## Prerequisite for all python based courses

FA-0260 · 3-day

Software Development

---

## Introduction

This 3-day course provides a comprehensive introduction to Python programming, designed specifically for beginners with no prior programming experience. Participants will learn the fundamental concepts of Python programming, including basic syntax, data structures, control structures, and how to write and debug simple Python programs. By the end of the course, learners will have a solid foundation in Python, which will prepare them for more advanced programming courses.

## Learning outcomes

Upon completion of this course, participants will be able to:

- Understand the basics of Python programming and its syntax
- Write and execute Python programs using various data types and operators
- Apply conditional statements and loops to control program flow
- Utilize built-in functions and create custom functions
- Manipulate Python data structures such as lists, tuples, and dictionaries
- Debug Python programs and handle common errors

## Prerequisites

No prior programming experience is required. However, a basic understanding of computer operations and familiarity with any operating system is beneficial.

## Metrics

Duration: 3 days

Timing: 9am to 5pm (with an hour's lunch break)

Assessments: in-class assessment

Type: Hands-on

Lab: Online lab

Software requirement:

- Web Browser
- Operating system with Administrative / Sudo / root access
- Terminal / console system
- VS Code

# Training outline

## 1. Day 1:

Introduction to Python Programming

- History and evolution of Python
- Features and applications of Python
- Python 2 vs. Python 3

Python Environment Setup

- Installing Python
- Using IDEs (Integrated Development Environments)
- Running Python scripts

Python Basics

- Variables and data types
- Operators (arithmetic, comparison, assignment, logical, etc.)
- Input and output functions
- Basic error handling

Control Structures

- Conditional statements (if, elif, else)
- Looping structures (for, while)
- Break and continue statements

## 2. Day 2:

Functions

- Defining and calling functions
- Function parameters and arguments
- Return statement and returning values
- Built-in functions

Data Structures

- Lists: creating, accessing, and modifying
- Tuples: creating, accessing, and modifying
- Dictionaries: creating, accessing, and modifying
- Common operations on data structures

File I/O

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

## 3. Day 3:

Exception Handling

- Understanding exceptions
- Try, except, and finally blocks
- Raising exceptions

Modules and Libraries

- Understanding modules and their importance
- Importing modules and using their functions
- Exploring commonly used Python libraries (e.g., math, random, datetime)

- Mini-Project
- Applying the concepts learned to create a simple Python application
- Debugging and error handling
- Next Steps and Resources
- Advanced Python programming topics
- Python frameworks and libraries
- Online resources and further learning opportunities

© 2026 Faysal Aziz · Kuala Lumpur