Technology + Knowledge = Innovation

Via Copenaghen, 10 Roma

www.vivasoft.it

info@vivasoft.it

Vivasoft

Consulting & Training

Microsoft

Vivasoft is a leading company in the technology sector, specializing in offering innovative solutions based on Microsoft. With years of experience in the market, we are proud to be certified Microsoft partners, committed to supporting the digital transformation of businesses. We provide a comprehensive range of Microsoft products and highly qualified training courses, designed to help companies optimize their processes, enhance productivity, and acquire advanced skills in the world of technology.

Module 1: Introduction to Python

What is Python and What is it Used For?

- Overview of Python and its applications
- Installing Python and setting up the development environment (IDE, text editors, Jupyter Notebook)
- Writing the first Python program ("Hello World")

Basic Python Syntax

- Variables, data types, and operators
- Handling user input and output
- Executing Python scripts
- Introduction to comments and documentation

Module 2: Fundamentals of Programming in Python

Data Types in Python

- Strings, numbers, lists, tuples, sets, and dictionaries

Operators and Control Structures

- Mathematical, logical, and comparison operators
- Control flow structures: if, else, elif, while, for

Functions and Exception Handling

- Defining functions, parameters, and return values
- Handling exceptions with try-except
- Introduction to modules and standard libraries

Module 3: Data Structures and Collections in Python

Working with Lists, Tuples, and Sets: Creating and modifying lists, Using tuples and sets effectively

Dictionaries and Advanced Data Manipulation:
creating, managing, and manipulating dictionaries

Sorting, Searching, and String Manipulation (Functions for sorting and searching collections)

Working with strings in Python

Technology + Knowledge = Innovation

Module 4: Object-Oriented Programming (OOP) with Python

What is object-oriented programming?
Classes and objects in Python
Attributes, methods, and constructors
Inheritance, polymorphism, and encapsulation
Special methods (__init__, __str__, __repr__)
Using abstract classes and interfaces
Handling custom exceptions

Module 5: Working with Files and Input/Output in Python

Basic file operations: opening, reading, and writing
Working with text and binary files
Handling CSV and JSON files
Using the os library for file and directory management
Exercises on data reading/writing and file manipulation
Working with Excel files using pandas

Module 6: Advanced Functions and Functional Programming in Python

Higher-order functions and lambda expressions
Decorators in Python
Generators and yield
Functional programming: map, filter, reduce
Functions as first-class objects
Using closures

Technology + Knowledge = Innovation

Module 7: Exception Handling and Logging

Advanced exception handling with try-except-finally
Creating custom exceptions
Using the logging module to track and record events
Logging to files and console
Debugging with tools like pdb

Module 8: Working with Python Libraries

Introduction to Python standard libraries
Working with datetime and calendar for date management
File manipulation with shutil and pathlib
Web interaction: using requests for APIs
HTML parsing with **BeautifulSoup**
Numerical and scientific computing with **NumPy** and **SciPy**
Data analysis with **pandas**

Module 9: Web Development with Python

Introduction to **Flask** and **Django**
Building a simple web application with Flask
Handling routes, views, and models in Flask
Working with SQL databases in Django
Authentication and authorization in Flask/Django
Creating a RESTful API with Flask
Introduction to templating and front-end technologies with Python

Technology + Knowledge = Innovation

Module 10: Testing and Debugging in Python

Writing unit tests with unittest and pytest
Creating integration and functional tests
Testing modules and functions
Debugging with pdb and pytest
Performance management and code profiling with cProfile
Simulating and mocking dependencies

Module 11: Scripting and Automation with Python

Writing scripts for file and directory management
Automating tasks with os and subprocess
Creating automatic backup scripts
Using **cron** to schedule tasks
Writing scripts to interact with web APIs
Automating cloud resource management

Module 12: Final Project

Developing a complete application (web, desktop, or automation)
Project design and planning
Writing code, testing, and debugging
Presentation and review of the final project

Technology + Knowledge = Innovation