Python Training – Essential Foundations for Data Science

Module 1 – Introduction to Python

- Overview of Python and its key applications in analytics and scientific computing
- Installing Python and choosing a development environment (IDLE, VS Code, or Jupyter)
- Getting started with the Python console and running your first scripts

Module 2 – Core Programming Concepts

- Understanding variables, data types, and type conversion
- Working with strings, numbers, and booleans
- Essential arithmetic and logical operations

Module 3 – Control Flow Essentials

- Writing conditional statements (if, elif, else)
- Using loops effectively (for, while)
- Mastering indentation and clean coding practices

Module 4 – Functions and Script Structure

- Defining and calling functions
- Passing arguments and returning values
- Organizing your code into clear, reusable blocks

Module 5 – Introduction to Object-Oriented Programming (Optional)

- Understanding objects and their role in Python
- Creating simple classes and instances

Module 6 – Working with Files

- Reading and writing text files
- Processing data line by line for real-world tasks

Module 7 – Practical Workshop

- Building a small end-to-end program combining variables, loops, and functions
- Personalized feedback and improvement tips

Master Git: Manage Your Versions and Collaborate Effectively

Module 1: Introduction to Git

Objective: Understand the usefulness of Git in version control and project development.

- Introduction to version control in the context of a static website project.
- Install and configure Git as well as Python, Pip, GitHub, GitLab, and MkDocs.
- Overview of Git and its ecosystem (GitHub, GitLab, Bitbucket, etc.).

Module 2: Working Alone with Git

Objective: Master the fundamentals of Git for an individual static website project using Python and Markdown.

- Basic commands: git init, git config.
- Track and record changes: git add, git status, git commit.
- Explore previous versions: git log, git diff.
- Manage versions: git tag.
- Publish online: git push.
- Discover MkDocs, some Python, and Markdown to run the site locally and host it online.
- Discover other concepts: branches (git branch, git checkout), a YAML file, a README.md, on GitHub and GitLab.
- Clone an online project: git clone.
- Work with command aliases.

Module 3: Working in a Team with Git

Objective: Learn how to use Git in a collaborative environment.

- Discover team types: owner and collaborator for a 2-person project, or equal collaborators.
- Manage teams and define permissions.
- Review commands and add updates to synchronize the local repository with the central repository: git fetch, git pull.
- Best practices for teamwork with Git.

Module 4: Pull Request (Merge Request)

Objective: Master the process of requesting code integration through a Pull Request.

- Introduction to Pull Requests, contributing to a project, and the code review / merge request process.
- Create and submit a Pull Request.
- Handle a conflict-free request: approve or reject the request.
- Update your repository: git fetch, git pull.
- Strategies for managing Pull Requests within a team.

Module 5: Managing Merge Conflicts

Objective: Learn how to handle and resolve version conflicts.

- Understand merge conflicts: causes and how to detect them.
- Handle a request with conflicts: proceed or reject the request.
- Resolve conflicts manually: git status, git mergetool.
- Use conflict resolution tools (VS Code, Meld, etc.).
- Best practices to avoid conflicts: use the stash git stash.
- Manage the stash: add, apply, drop, and more.

Module 6: Synchronization and Continuous Integration

Objective: Manage team integration workflows.

- Differentiate between origin and upstream repositories.
- Introduction to CI/CD concepts, continuous integration, and triangular workflows.
- Introduction to branches: git branch, git checkout -b.
- Introduction to HTTP and SSH remotes and managing SSH encryption keys.
- Discover backtracking, history, and other user-friendly tools.
- Explore the CI/CD pipeline, continuous integration, and continuous deployment.

Training: Adobe Premiere Pro

Introduction to the Adobe Premiere Pro Training

- Introduction to non-linear editing
- Interface: Panels and Tools
- Workspaces

Capture and Project Settings

- Standard and High Definition
- Create and edit project settings in Premiere Pro
- Adjust user preferences
- Capture video and audio
- Recording and capture settings
- Batch capture

Importing Assets into Adobe Premiere Pro

- Manage media in the Project panel
- Manage media in the Media Browser
- Overview of formats: video, audio, images
- Video, audio, image and graphic issues
- Using the Source panel
- Display and recall options

Adobe Premiere Pro Editing Basics

- Set In and Out points
- Overview of editing and sequences
- Add or remove audio/video tracks
- Create a first edit
- Basic video editing tools and markers
- Program and reference monitors
- Trim audio and video clips
- Work with the Ripple Edit tool and the Ripple Delete tool
- Three-point editing

Effects and Transitions

- Effects and Effect Controls panel
- Introduction to audio and video transitions
- Add, remove, and edit transitions
- Use the Razor tool and editing tools for transitions

Audio and Video Filters

- Basic video and audio enhancement filters
- Understanding video color correction
- Apply noise reduction for audio and video
- Basic color correction

Creating Titles

- Work with the Title window
- Titling tools: actions, properties
- Lettering styles
- Use titling templates

- Use keyframes

Add / Remove Keyframes

- Edit keyframes
- Audio tracks with the Audio Mixer panel
- Exporting

Exporting: Adobe Media Encoder

- Formats: QuickTime, H.264
- Windows Media: Adobe Encore
- Export settings for YouTube, Vimeo, Apple TV and portable media players
- Export to SD DVD or Blu-Ray

Conclusion of the Premiere Pro Training

---

quote formation

AI: Evolution, Understanding, Application, and Programming

Module 1: Introduction to Artificial Intelligence

- Overview and Q&A session (1h30)
- Definition and evolution of AI:
  - Terminology and evolution
  - From early algorithms to neural networks
  - Deep Learning and Generative AI
  - Weak, General, and Superintelligent AI
  - Examples of applications across sectors
- Factors supporting AI:
  - Algorithmic and heuristic structures
  - Hardware aspects (CPU, GPU, TPU)
  - Languages (R, Python, C++, Rust, Mojo)
  - Software and frameworks
- Introduction to Machine Learning:
  - Main model categories
  - Supervised and unsupervised learning
  - Applications and use cases
- Introduction to Deep Learning:
  - Differences from Machine Learning
  - Concepts and models
  - Applications and use cases
- Natural Language Processing features
- Overview of Generative AI:
  - Language, image, and multimodal models
  - Market applications and products
  - Example: Google Gemini
- Issues and risks:
  - Ethics and bias
  - Privacy and environmental impact
  - Risk of dystopian outcomes

Module 2: AI Demonstration

- Overview and Q&A session (1h30)
- DevOps and AI pipelines:
  - Data preprocessing
  - Training and optimization
  - Deployment
- Machine Learning in practice:

    - Demo with Scikit-Learn
    - User-friendly approach with PyCaret
    - Note: Source code via Google Colab
- Deep Learning in practice:
    - Demo with Numpy
    - Using PyTorch
    - Note: Source code via Google Colab
- Generative AI:
    - Exploring large language and multimodal models
    - Fine-tuning and autonomous agents
    - Note: Hands-on in the next module with Google Gemini

Module 3: Using Google Gemini

- Hands-on experimentation with Google Colab (3h30)
- Using the chatbox:
    - Crafting simple prompts
    - Tips for Python and other technical prompts
- Using the Python API:
    - Designing advanced prompts
    - Code optimization and unit testing
- Using Gemini Studio:
    - Fine-tuning and Retrieval-Augmented Generation
    - Building an autonomous agent
    - Experimental projects with Python and more

Follow-up training with OpenAI

OpenAI Training for Developers

Planner Training: Getting Started with Microsoft Planner

Module 1: Introduction to Microsoft Planner

- Role of Planner within the Microsoft 365 ecosystem
- Access via browser, Teams, and mobile app
- Typical use cases: simple task management, team projects, routine tracking

Module 2: Creating a Plan

- Create a new plan via Planner or Teams

- Choose or create a Microsoft 365 group
- Understand the structure: buckets and tasks

Module 3: Task Management

- Create and edit a task
- Assign a task owner
- Add: due date, priority, checklist, attachments, notes
- Track progress (Not started, In progress, Completed)

Module 4: Visual Organization

- Create buckets to structure project phases
- Use color-coded labels
- Rearrange tasks via drag-and-drop in Board view

Module 5: Basic Collaboration

- Comment on tasks
- Understand automatic notifications
- Integrate Planner into Microsoft Teams (tab, simplified view)

Module 6: Guided Hands-on Practice

- Create a mini project in pairs or groups
- Apply the features covered
- Open discussion and Q&A

Shopify Training

Module 1 – Introduction to E-commerce and Shopify

- Understand how e-commerce works (B2C, B2B, Dropshipping)
- Overview of **Shopify** and its ecosystem
- Create an account and complete the initial setup
- Tour of the Shopify dashboard (Shopify admin)
- Quick comparison with WordPress + WooCommerce

Module 2 – General Store Configuration

- General settings (language, currency, time zone)
- Payment configuration (Shopify Payments, PayPal, Stripe)
- Tax setup (Canada / Quebec)
- Shipping rates and delivery zones configuration
- Writing policies (returns, privacy, terms of use)

Module 3 – Product and Collection Management

- Create and edit products
- Manage variants (size, color, model)
- Organize collections (manual and automatic)
- Inventory and stock management

Module 4 – Website Design and Customization

- Select and install a theme
- Customize colors, fonts, and sections
- Create the homepage
- Manage menus and navigation
- Mobile optimization (responsive design)

Module 5 – Essential Applications

- Overview of the Shopify App Store
- Install a useful app (e.g., customer reviews or shipping)
- Manage and remove applications
- Best practices to avoid unnecessary apps

Module 6 – Visibility and Promotions Basics

- Basic SEO settings (title and description)
- Create a simple promotional code
- Connect social media accounts
- Review key statistics

Module 7 – Order Management and Performance Tracking

- Manage and process orders
- Refunds and cancellations
- Payment tracking
- Sales reports and statistics
- Analyze key performance indicators (KPIs)

Training: WPF Interface Development with C# and Visual Studio

Module 1 – Introduction and WPF Fundamentals

- Overview of WPF and comparison with Windows Forms

- WPF architecture: XAML, Code-Behind, Data Binding
- Creating your first WPF project in Visual Studio
- Understanding XAML and its relationship with C#
- Using core controls (Button, TextBox, Label, ListBox, etc.)
- Organizing the interface with layout containers (Grid, StackPanel, DockPanel, WrapPanel)

Module 2 – Styling and Layout

- Managing resources (Resources, ResourceDictionaries)
- Introduction to styles (Styles, Control Templates)
- Applying themes and customizing controls
- Using layouts to build flexible user interfaces
- Introduction to event handling in WPF
- Hands-on exercise: building a cohesive mini user interface

Module 3 – Data Binding and MVVM

- In-depth understanding of DataContext and Binding
- Binding modes (One-Way, Two-Way, etc.)
- Working with lists and collections (ItemsControl, ListView, DataGrid)
- Data validation and conversion (ValueConverters)
- Introduction to the MVVM pattern (Model-View-ViewModel)
- Practical case study: implementing a basic MVVM architecture

Module 4 – Practical Project and Advanced Features

- Navigation between views (UserControl, Pages, Navigation)
- Command handling (Commands, RelayCommand)
- Interacting with a database or service (simple example using Entity Framework or a mock API)
- Managing multimedia resources (images, icons)
- Best practices for structuring a WPF project
- Final workshop: developing a complete interface aligned with your C# project

---

PHP Training: The Most Comprehensive Course

Introduction to PHP Training

Learning the Basics of PHP

Why Use PHP?
Structure of a PHP Page
Embedding PHP Code in an HTML Document
Variable Types
Declaring and Using Arrays
Constants
Operators
Loops and Conditional Statements
Superglobal Variables
Retrieving Form Data
Sending Emails

Hands-on Workshop: *Review of core HTML tags and styling commands, setting up a template page*

Creating Functions and Using Built-in Functions

Declaring Functions
Parameters and the Return Statement
Creating Function Libraries
Built-in Functions: Examples, Organization, Documentation
Functions for Dates, Arrays, and Strings
Include and Require Functions

Hands-on Workshop: *Creating custom functions and using built-in functions*

Processing Form Data

Creating a Form
Retrieving Form Data
File Uploads: The $_FILES Variable and Related Functions
Sending Emails

Hands-on Workshop: *Building a job application form*

Interacting with Users

Retrieving and Using GET Parameters
Using Cookies to Store Information
File Uploads: The $_FILES Variable and Related Functions
Sending Emails

Hands-on Workshop: *Building a job application form*

phpMyAdmin

Overview of phpMyAdmin

Creating a Database with phpMyAdmin
Creating Tables and Exploring Field Types
Modifying Table Structure

Hands-on Workshop: *Creating a MySQL database and tables*

SQL

Selecting Records (SELECT)
Inserting Records (INSERT INTO)
Updating Records (UPDATE)
Deleting Records (DELETE)
Operators and Conditions

Hands-on Workshop: *Writing basic SQL queries*

Using a MySQL Database with PHP

Overview of Database Management Systems
Data Types
Connecting to a MySQL Server
Populating the Database from a Form
Inserting Data into the Database with PHP
Updating Data with PHP
Deleting Data with PHP
Securing Data Before Sending to MySQL

Hands-on Workshop: *Storing job application form data in the database*

Cookies & Sessions

Session Overview
Advantages and Disadvantages of Cookies and Sessions
Storing and Retrieving Session Variables
Storing and Retrieving Cookie Variables

Hands-on Workshop: *Building a simple shopping cart and user interface*

Working with Files

Opening and Writing Files

SQL: Advanced Concepts

Tables and Relationships
Joins: INNER, LEFT, and RIGHT JOIN

Hands-on Workshop: *Writing SQL queries across multiple tables*

Building a Mini CMS

CMS Concepts
Existing CMS Platforms
Introduction to Data Modeling
Creating the Public Interface
Creating the Admin Interface
User Management

Hands-on Workshop: *Library management application*

XML

Overview of the XML Format
Writing an XML Document
Introduction to XML

Hands-on Workshop: *Creating an RSS feed and generating a sitemap for the library project*

Java Training: The Most Complete Training

Introduction to Java Training

Establishment of the development environment
Java programming syntax

Structure of a .java file

Primitive type variable and operator

Comment and document code

The conditions in the code: if / else, ….

Comparison and logic operators

The tables

Buckles

Class method

Object-oriented programming

Goal

OOP comparison and procedural language

Class and object

Methods and attributes

Relationships between classes

Interface concept

Abstract class

Practical Workshop: Class Designs and Linking Classes

Exceptions in the Java language

Principle of an exception

Manipulate exceptions

Java input / output management

Package java.io

Operator reading / writing

Format: binary, text

Character set and encoding

Collections & Files

Presentation of the collections

Iterate on collections

Log Management: java.util.logging

Reading and writing files

Using buffers

Practical workshop: Manipulation of object collections

Java and databases

Principle of the JDBC API

Connect to a database

Retrieve data from a database

Link database data with Java objects

Insert data into the database