

# Python

## Introduction to Python

- Introduction to Python and its applications.
- Setting up Python and IDEs (Jupyter Notebook, VS Code, PyCharm).
- Python Basics:
  - Variables, data types, and typecasting.
  - Input/Output, comments, and code structure.

## Hands-On Practice:

- Write simple Python scripts to perform basic operations.
- 

## Control Flow and Functions

- Conditional Statements: if, elif, and else.
- Loops: for, while, and nested loops.
- Functions in Python:
  - Defining and calling functions.
  - Arguments, parameters, and return values.
  - Lambda functions.

## Hands-On Practice:

- Create a calculator program with conditional logic and reusable functions.
- 

## Data Structures and File Handling

- Python Data Structures:
  - Lists, tuples, sets, and dictionaries.
  - List comprehension and dictionary comprehension.
- File Handling: Reading, writing, and appending files.
- Exception Handling: try, except, finally.

## Hands-On Practice:

- Process text files to count word frequencies and handle exceptions.
-

## Working with Python Libraries

- Introduction to Libraries and Modules.
- NumPy for Numerical Computation:
  - Arrays, operations, and basic statistics.
- Pandas for Data Manipulation:
  - DataFrames, indexing, and data cleaning.
- Matplotlib and Seaborn for Data Visualization:
  - Creating plots and visualizing trends.

### Hands-On Practice:

- Analyze a dataset using Pandas and create visualizations.
- 

## Python for Data Science

- Exploratory Data Analysis (EDA) with Pandas.
- Machine Learning Basics with Scikit-learn:
  - Regression and classification models.
  - Model evaluation metrics.
- Introduction to TensorFlow/PyTorch for Deep Learning (Optional for advanced learners).

### Hands-On Practice:

- Build and evaluate a simple predictive model using Scikit-learn.
- 

## Capstone Project

- Build an end-to-end project integrating:
  - Python programming fundamentals.
  - Data manipulation and visualization.
  - Web development or automation.

### Examples of Projects:

1. E-commerce Dashboard: Analyze and visualize sales data
  2. Task Automation: Automate email reminders for deadlines.
  3. Personal Portfolio Website: Built with Flask/Django.
- 

## Assessment Structure

- Weekly assignments and quizzes.
  - Mid-course project submission.
  - Final capstone project with peer review and feedback.
- 

### **Tools and Resources**

- Python Installation: [python.org](https://python.org)
- Libraries: NumPy, Pandas, Matplotlib, Seaborn, Flask, Scikit-learn.
- Dataset Repositories: Kaggle, UCI Machine Learning Repository.