Python for Advanced Excel Users

Duration: 3 days
Prerequisites: Knowledge of Python Programming

Day 1 – Transitioning from Excel to Python

Topics:

1. Why Python for Excel Users
    - When Excel reaches its limits
    - Python advantages: automation, big data, advanced analysis
2. Python Basics in Excel Context
    - Variables, data types, and data structures (list, dict, DataFrame)
    - Python equivalents for Excel formulas (SUM, IF, VLOOKUP, etc.)
3. Working with Excel Files in Python
    - Reading/writing Excel with openpyxl and pandas
    - Multiple sheets, ranges, and formats
4. Data Cleaning & Preparation
    - Removing duplicates, handling missing values
    - Text cleaning (case, trim, split, regex)
    - Date/time handling and transformations

Labs:

- Import multiple Excel reports and merge into one dataset
- Replicate VLOOKUP & PivotTable logic in Python using pandas

---

Day 2 – Advanced Analysis & Automation

Topics:

1. Advanced Data Manipulation
    - GroupBy & aggregation for business summaries
    - Calculated fields & KPI creation
    - Cross-tabulations (pivot-like operations)
2. Automating Excel Tasks
    - Generating multiple reports automatically
    - Conditional formatting via Python
    - Automating repetitive monthly/weekly report generation
3. Visualization for Excel Users
    - Matplotlib, Seaborn basics
    - Replacing Excel charts with Python plots
    - Embedding charts back into Excel files
4. Large Dataset Handling
    - Processing files larger than Excel's limit
    - Combining and analyzing multi-year datasets

Labs:

- Automate monthly sales report with multiple KPIs and charts in one Excel file
- Create a Python script that refreshes and formats data from multiple sources

---

Day 3 – Dashboards, Integration & Capstone

Topics:

1. Building Interactive Dashboards
    - Streamlit for web-based dashboards
    - Adding filters, dropdowns, and live charts
2. Excel-Python Integration
    - Running Python directly inside Excel (using xlwings)
    - Calling Python scripts from Excel buttons
3. Connecting to External Data Sources
    - SQL databases & APIs for dynamic data
    - Merging external data into Excel dashboards
4. Best Practices for Excel-to-Python Workflows
    - File organization, error handling
    - Scheduling automated report generation

Capstone Project:

- **Scenario:** Automate a **Finance KPI Dashboard**
    - Pull financial data from Excel & SQL
    - Clean & transform the data
    - Generate charts & KPIs automatically
    - Embed results in both Excel and a web dashboard