Python: Data Handling, Visualization & API Development

Duration: 8 days
Prerequisites: Working Knowledge of Python Programming

Day 1 – Core Python: Best Practices & Debugging

- Control structures (quick recap – if/else, loops)
- Functions & scope (advanced use)
- Error types & debugging basics
- String manipulations: regex, .split(), .replace(), .strip()
- Using logging instead of print
- Secure secrets with os.environ & .env (python-dotenv)
- Performance optimization: generators, iterators, memory efficiency
  Lab: Secure API script with logging, regex parsing & error handling

---

Day 2 – Collections & Data Handling with pandas

- Lists, Tuples, Sets – advanced operations
- Dictionaries – nested & complex structures
- File Handling (CSV, JSON, Excel)
- pandas: Series, DataFrame, reading/writing files
- pandas operations: filtering, grouping, aggregation
  Lab: Clean a dataset & create summary reports with pandas

Day 3 – Data Visualization & Reporting

- matplotlib & seaborn basics
- Line, bar, histogram, scatter plots
- Customizing & exporting visualizations
- Automating report generation (Excel, PDF, HTML)
  Lab: Sales analytics dashboard with pandas + matplotlib

Day 4 – Automation & Integration

- Web Scraping with requests & BeautifulSoup
- API Handling (REST APIs, authentication, requests library)
- Task Automation with os, shutil, schedule
- Automating repetitive Excel/CSV reports
  Lab: Automated daily report generator with API + visualization

---

Day 5 – Advanced Python & Modern Development

- OOP (classes, inheritance, polymorphism)
- Exception handling advanced patterns
- Virtual environments & dependency management

- Git & GitHub integration (commits, push, branching)
- GitHub Copilot for Python coding assistance
- Creating reusable Python packages
  Lab: Package a Python utility & push to GitHub

Day 6 – Building Interactive Data Apps

- Introduction to Streamlit
- Interactive dashboards with input widgets & filters
- Dynamic updates & visualization integration
- Deploying Streamlit apps (Streamlit Cloud / Heroku)
  Lab: Build & deploy a Streamlit dashboard (COVID-19 tracker / Sales report)

Day 7 – Django REST Framework (Intro & Setup)

- Django & DRF architecture
- Setting up Django project & apps
- Models, migrations, database setup
- Serializers & Views in DRF
- Building basic CRUD APIs
  Lab: Student Management API with DRF (CRUD operations)

Day 8 – Django REST Framework (Advanced) + Capstone

- Authentication (Token, JWT basics)
- ViewSets, Routers, Pagination
- API versioning & testing with Postman
- Securing APIs (auth & permissions)

- **Capstone Project:**
  - Data collection (web/API)
  - Clean & visualize with pandas + matplotlib
  - Build interactive Streamlit dashboard
  - Expose REST API with Django REST Framework
  - Push project to GitHub