60-Day Python Roadmap – Inventory Management System Capstone

# 🎯 Overview

This roadmap builds on your 30-Day Python Challenge to advance your skills with:  
✅ Deeper understanding of Python fundamentals  
✅ Real-world libraries (requests, pandas, sqlite3, etc.)  
✅ Automation scripting  
✅ A full Inventory Management System project

# 🟢 Phase 1: Core Python Refresher (Days 1–10)

Topics to Cover:  
- Advanced Data Structures  
- Comprehensions  
- Functions and Lambdas  
- Exception Handling  
- OOP  
- File I/O (JSON/CSV)

|  |  |  |
| --- | --- | --- |
| Day | Focus | Task |
| 1 | Data Types | Revise lists, dicts, tuples, sets |
| 2 | Loops & Comprehensions | Practice nested loops, list comprehensions |
| 3 | Functions & Lambdas | Build mini calculator functions |
| 4 | Exception Handling | Write code with try/except, custom exceptions |
| 5 | OOP Basics | Create a `Product` and `Category` class |
| 6 | OOP Inheritance | Extend `Item` into `PerishableItem` and `NonPerishableItem` |
| 7 | Magic Methods | Implement `\_\_str\_\_`, `\_\_len\_\_`, etc. |
| 8 | File I/O Basics | Read/write text and CSV files |
| 9 | JSON Handling | Parse and save JSON data |
| 10 | Mini Project | Simple Contact Book (CRUD in CSV) |

# 🟡 Phase 2: Libraries & Intermediate Projects (Days 11–30)

Topics:  
- requests  
- BeautifulSoup  
- pandas  
- sqlite3  
- argparse

|  |  |  |
| --- | --- | --- |
| Day | Focus | Task |
| 11 | requests Basics | Fetch JSON data from a public API |
| 12 | requests Advanced | Handle headers, params, save response to JSON file |
| 13 | BeautifulSoup Basics | Parse HTML and extract elements |
| 14 | Web Scraping Project | Scrape sample quotes or products |
| 15 | pandas Basics | Read CSV, clean data, calculate summary statistics |
| 16 | pandas Advanced | GroupBy, pivot tables |
| 17 | sqlite3 Basics | Create a database and a table |
| 18 | sqlite3 CRUD | Insert, update, delete records |
| 19 | argparse Basics | Build a CLI utility |
| 20 | Expense Tracker | CRUD with SQLite and CLI interface |
| 21 | pandas/sqlite3 Deep Dive | Enhance Expense Tracker |
| 22 | pandas/sqlite3 Deep Dive | Enhance Expense Tracker |
| 23 | pandas/sqlite3 Deep Dive | Enhance Expense Tracker |
| 24 | Project Refactoring | Clean up code, improve error handling |
| 25 | Project Refactoring | Clean up code, improve error handling |
| 26 | Extra Practice | Mini projects: Weather App, JSON parser |
| 27 | Extra Practice | Mini projects: Weather App, JSON parser |
| 28 | Extra Practice | Mini projects: Weather App, JSON parser |
| 29 | Recap & Notes | Summarize learnings, prepare reference cheatsheets |
| 30 | Recap & Notes | Summarize learnings, prepare reference cheatsheets |

# 🟣 Phase 3: Automation & Real-World Use Cases (Days 31–45)

Topics:  
- File/folder automation (os, shutil, pathlib)  
- Sending emails (smtplib)  
- Scheduling tasks (schedule)  
- Excel reports (openpyxl)

|  |  |  |
| --- | --- | --- |
| Day | Focus | Task |
| 31 | os & shutil | Move, rename, and delete files |
| 32 | pathlib | Path manipulations |
| 33 | smtplib Email | Send test emails from Python |
| 34 | Automate Reports | Email daily reports |
| 35 | schedule Library | Schedule script execution |
| 36 | File Organizer Project | Auto-sort files into folders |
| 37 | File Organizer Project | Continue implementation |
| 38 | File Organizer Project | Finish and test |
| 39 | Email Sender Project | Automated email reports |
| 40 | Email Sender Project | Continue implementation |
| 41 | Email Sender Project | Finish and test |
| 42 | Excel Report Generator | Generate Excel reports |
| 43 | Excel Report Generator | Continue implementation |
| 44 | Excel Report Generator | Finish and test |
| 45 | Documentation | Summarize and document all scripts |

# 🔵 Phase 4: Capstone Project – Inventory Management System (Days 46–60)

Project Overview:  
A console-based Inventory Management System to:  
✅ Add, update, delete products  
✅ Track quantities and categories  
✅ Save data to SQLite  
✅ Optional: Build a Tkinter GUI

|  |  |  |
| --- | --- | --- |
| Day | Focus | Task |
| 46 | Planning & Design | Draw ER diagram, plan database schema |
| 47 | Planning & Design | Define models and workflow |
| 48 | Planning & Design | Set up initial project structure |
| 49 | Database Setup | Create tables, set up connection module |
| 50 | Database Setup | Seed sample data |
| 51 | Database Setup | Test database functions |
| 52 | Core Logic | Implement CRUD functions |
| 53 | Core Logic | Continue CRUD implementation |
| 54 | Core Logic | Test and debug core logic |
| 55 | User Interface | Build CLI menus and prompts |
| 56 | User Interface | Refine user experience |
| 57 | User Interface | Integrate with database |
| 58 | Testing & Debugging | Test all flows and handle edge cases |
| 59 | Testing & Debugging | Polish and finalize code |
| 60 | Finalization | Prepare README, push to GitHub |

# 🔗 Recommended Tutorials & Docs

- Real Python: https://realpython.com  
- Automate the Boring Stuff: https://automatetheboringstuff.com  
- SQLite Tutorial: https://www.sqlitetutorial.net/sqlite-python/  
- Tkinter Guide: https://realpython.com/python-gui-tkinter/  
- FastAPI: https://fastapi.tiangolo.com

# 🗃️ Suggested GitHub Repository Structure

inventory-management-system/  
├── main.py  
├── database.py  
├── models.py  
├── cli.py  
├── utils.py  
├── README.md  
└── requirements.txt