

Python Roadmap (Beginner to Advanced)

1. PYTHON FOUNDATIONS (*Beginner Core*)

- Variables, data types, type casting
- Control flow (if, for, while, break, continue)
- Functions, return values, args & kwargs, lambda functions
- Error handling (try/except/finally)
- Virtual environments (venv)

2. DATA STRUCTURES + CORE TECHNIQUES

- Lists, tuples, sets, dictionaries
- List/dict comprehensions
- String manipulation & regular expressions (basics)
- Sorting & searching algorithms

3. OBJECT-ORIENTED PROGRAMMING (Intermediate)

- Classes, objects, constructors
- Inheritance, encapsulation, polymorphism
- Magic/dunder methods
- Decorators & property decorators

4. FILE HANDLING + AUTOMATION

- File reading/writing (CSV, JSON, text files)
- Testing with pytest (unit tests, assertions)
- Working with APIs (requests library)
- Web scraping (BeautifulSoup, Selenium)
- Basic CLI tools (argparse)

5. DATABASES (SQL + ORM)

- SQL fundamentals (CRUD operations, joins, filtering)
- SQLite, PostgreSQL
- SQLAlchemy ORM
- Database design principles

6. VERSION CONTROL + COLLABORATION

- Git basics (add, commit, push, pull)
- Branching, merging, pull requests
- GitHub workflows

- README files & project documentation

7. WEB DEVELOPMENT

- HTML/CSS
- Flask (routes, templates, forms, sessions)
- Authentication & authorization
- REST APIs
- FastAPI
- Deployment (Render, Railway, Vercel)

8. ASYNC PROGRAMMING

- asyncio fundamentals as sync/await syntax
- Concurrent vs parallel execution
- Practical use cases (APIs, web scraping)

9. DATA ANALYSIS & VISUALIZATION

- Pandas (DataFrames, cleaning, transformation)
- Exploratory Data Analysis (EDA)
- Matplotlib, Seaborn, Plotly
- Working with real datasets

10. MACHINE LEARNING FUNDAMENTALS

- Scikit-learn basics
- Supervised learning (classification, regression)
- Model evaluation (train/test split, metrics)
- Random Forest, decision trees
- Feature engineering basics

11. ALGORITHMS & COMPUTER SCIENCE

- Big-O notation & complexity analysis
- Common algorithms (binary search, sorting algorithms)
- Data structures (stacks, queues, hash tables)
- Problem-solving patterns

12. PROJECT PHASE (Portfolio Building)

Beginner Projects:

- CLI calculator or to-do app
- Web scraper for specific data

- Simple REST API

Intermediate Projects:

- Full-stack Flask/FastAPI app with database
- Automation script (email sender, file organizer)
- Data analysis project with visualizations

Advanced Projects

- ML model with Flask API deployment
- Multi-feature web application
- Open-source contribution

13. **DEPLOYMENT & DEVOPS ESSENTIALS**

- Docker fundamentals (containers, images, Dockerfile)
- Environment variables & secrets management
- CI/CD basics (GitHub Actions)
- Cloud platforms (AWS/GCP basics)Monitoring & logging

Prepared by: [Benard Joseph Tony](#)