Python Learning Syllabus: 0 to Expert Level

Stage 0: Foundations (0-2 weeks)

- Introduction to programming
- Installing Python and setting up environment (IDEs, Jupyter, VS Code)
- Running Python scripts
- Basic syntax: indentation, comments, variables
- Data types: integers, floats, strings, booleans
- Basic input/output
- Simple arithmetic operations
- Type casting

Stage 1: Core Python (2-4 weeks)

- Strings: indexing, slicing, methods
- Lists: creation, indexing, slicing, methods
- Tuples: immutable sequences, unpacking
- Sets: creation, operations
- Dictionaries: key-value pairs, methods
- Conditional statements: if, elif, else
- Loops: for, while, break, continue
- Functions: defining, arguments, return values, scope
- Modules and importing

Stage 2: Intermediate Python (4-6 weeks)

- List comprehensions
- Dictionary and set comprehensions
- Lambda functions
- Map, filter, reduce
- Exception handling: try, except, finally, raising errors
- File handling: open, read, write, with context

- Regular expressions basics
- Built-in modules: math, datetime, random, os, sys

Stage 3: Object-Oriented Python (3-5 weeks)

- Classes and objects
- Instance variables and methods
- Class variables and methods
- Inheritance and polymorphism
- Encapsulation and private attributes
- Special methods: **init**, **str**, **repr**, **len**

Stage 4: Advanced Python Concepts (4-6 weeks)

- Iterators and generators
- Decorators
- Context managers and with statement
- Advanced file handling (csv, json)
- Logging
- Functional programming concepts
- Python memory management and garbage collection

Stage 5: Data Handling and Libraries (6-8 weeks)

- Numpy: arrays, vectorized operations, indexing
- Pandas: Series, DataFrames, filtering, aggregation
- Matplotlib/Seaborn: basic plotting, customization
- Working with APIs: requests, JSON parsing
- Web scraping basics: BeautifulSoup, Selenium

Stage 6: Testing, Packaging, and Deployment (3-5 weeks)

- Unit testing with unittest or pytest
- Debugging techniques
- Virtual environments and pip

- Packaging Python projects
- Creating and running scripts
- Introduction to version control with Git

Stage 7: Advanced Projects and Specialization (6-10 weeks)

- Object serialization (pickle, json)
- Multi-threading and multiprocessing
- Async programming: asyncio, async/await
- Database connectivity: SQLite, SQLAlchemy
- Building small projects: web scraper, REST API, automation scripts
- Optional specialization:
- Data Science: pandas, numpy, matplotlib, scikit-learn
- Web Development: Flask, Django
- Automation: Selenium, PyAutoGUI
- Game Development: Pygame

Stage 8: Expert Level (ongoing)

- Design patterns in Python
- Performance optimization
- Memory profiling
- Advanced OOP and metaclasses
- Custom decorators and context managers
- Contributing to open source
- Large-scale project development
- Code quality and PEP8 compliance