

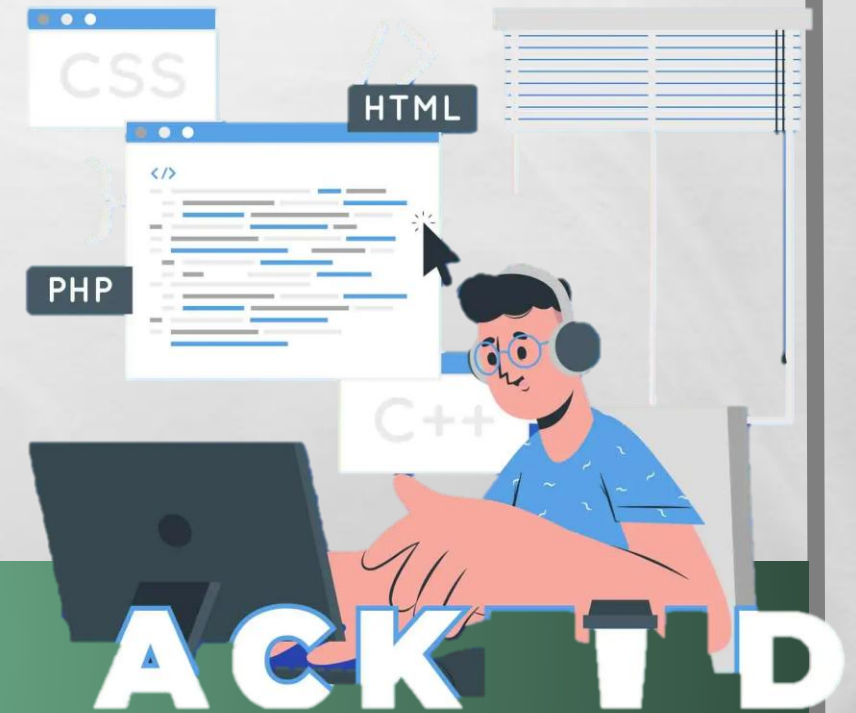
PYTHON BACKEND DEVELOPER

PRESENTED BY SWASTIK SOMPURA



WHAT IS BACKEND DEVELOPMENT

- **BACKEND DEVELOPMENT IS THE PART OF A WEBSITE OR APP THAT USERS DON'T SEE. IT'S LIKE THE "BEHIND THE SCENES" WORK THAT MAKES EVERYTHING FUNCTION PROPERLY.**
- **LIKE NODE.JS, PYTHON, JAVA, RUBY, AND DATABASES SUCH AS MYSQL, POSTGRESQL, AND MONGODB.**



ACK ID

PYTHON PAST

- 1990s – Python introduced by **Guido van Rossum**.
- Early backend work used **CGI scripts** for web servers.
- Limited frameworks, slower performance.
- Mainly used for **prototyping and scripting**.

PYTHON GROWTH (2000S–2010S)

- RISE OF WEB FRAMEWORKS LIKE DJANGO (2005) AND FLASK (2010).
- ADOPTION IN STARTUPS AND ENTERPRISES FOR BACKEND SYSTEMS.
- INTEGRATION WITH DATABASES (MYSQL, POSTGRESQL, SQLITE).
- POPULAR FOR REST APIS, CMS, AND E-COMMERCE.

PYTHON PRESENT

- WIDELY USED IN BACKEND WEB DEVELOPMENT.
- MODERN FRAMEWORKS: FASTAPI, DJANGO, FLASK.
- USED FOR MICROSERVICES, APIS, AND CLOUD-NATIVE APPS.
- INTEGRATION WITH AI/ML, IOT, AND DATA SCIENCE.
- SUPPORTED BY CLOUD PLATFORMS (AWS, AZURE, GCP).

PYTHON FUTURE

- AI-POWERED BACKENDS (CHATBOTS, RECOMMENDATION ENGINES).
- GROWTH OF ASYNCHRONOUS FRAMEWORKS (FASTAPI, AIOHTTP).
- STRONG ROLE IN SERVERLESS COMPUTING AND EDGE COMPUTING.
- ENHANCED PERFORMANCE WITH PYPY, CYTHON, RUST + PYTHON INTEGRATION.
- CONTINUED DOMINANCE IN DATA + BACKEND HYBRID APPLICATIONS.

Python



Roadmap

Step 1: Learn the Basics- Syntax, Variables, Data Types, Conditionals,

Step 3: Data Structures- Lists, Tuples, Sets, Dictionaries

Step 5: Advance Topics 1- RegEx, Decorators, Lambda

Step 7: Learn Python Libraries

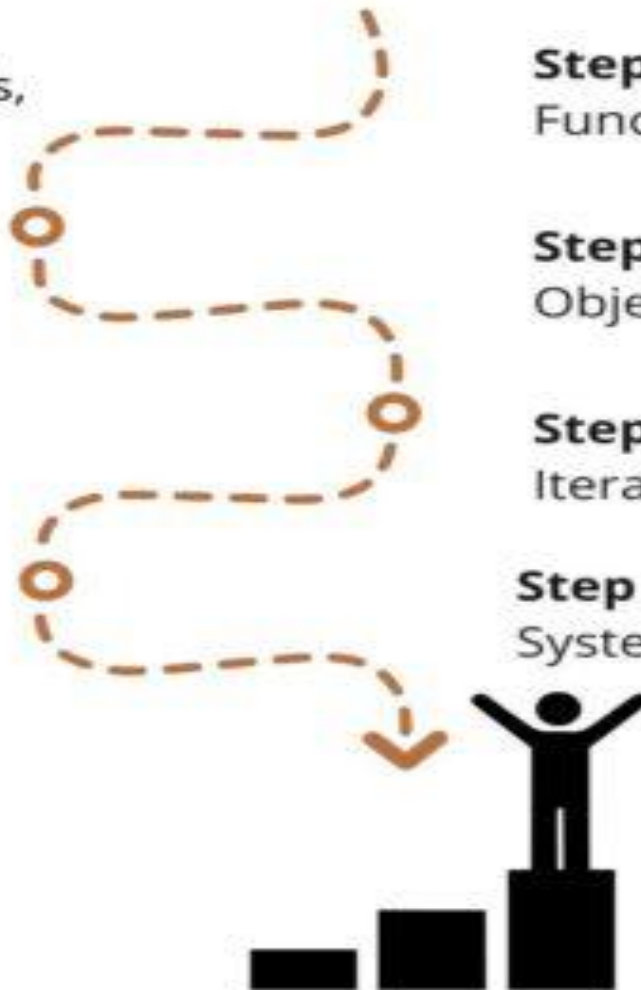
Step 9: Build Python Apps

Step 2: Loops, Functions, Built-in Functions

Step 4: OOP- Classes, Inheritance, Objects

Step 6: Advanced Topics 2- Modules, Iterators,

Step 8: Learn Version Control Systems



THANK YOU