

Python Backend Development with Postgres SQL

Duration: 5 days / 40 Hours

Prerequisites: Knowledge of Python and API Development

Day 1: Python Programming Basics

Module 1: Python Introduction

- Installing Python and IDE Setup
- Python Syntax, Variables, and Data Types
- Input/Output, Type Casting

Module 2: Control Structures & Functions

- Conditional Statements
- Loops – for, while
- Functions and Parameters
- Lambda Functions

Module 3: Data Structures in Python

- Lists, Tuples, Sets, Dictionaries
- List Comprehensions
- Iteration and Built-in Functions

Labs:

- Write simple Python programs
 - Create a calculator and a basic dictionary app
-

Day 2: OOP & Introduction to Web Development

Module 4: Object-Oriented Programming (OOP)

- Classes and Objects
- Constructors, Instance and Class Variables
- Inheritance and Polymorphism

Module 5: Web Development Basics

- What is Backend?
- Web Architecture: Client-Server, HTTP Basics
- Introduction to Flask
- Creating First Web App with Flask

Labs:

- Create and test a class-based Python app
 - Set up a basic Flask Hello World API
-

Day 3: REST APIs and PostgreSQL

Module 6: Building REST APIs

- Routes and Views (GET, POST, PUT, DELETE)
- Request Parameters and JSON Handling
- Error Handling & Status Codes

Module 7: Introduction to PostgreSQL

- Installing PostgreSQL & pgAdmin
- Basic SQL Commands: CREATE, SELECT, INSERT, UPDATE, DELETE
- Data Types and Constraints

Labs:

- Build a simple REST API (e.g., Book Inventory)
 - Create and manipulate a database in PostgreSQL
-

Day 4: Python-PostgreSQL Integration

Module 8: Database Connectivity with Python

- psycopg2: Connecting and Executing Queries
- SQLAlchemy ORM: Models and Sessions
- Creating Tables from Python

Module 9: Complete API with Database

- Integrating Flask/FastAPI with PostgreSQL
- Create, Read, Update, Delete Records via API
- API Testing with Postman

Labs:

- Build and test API endpoints for a "Student Management System"
 - Use SQLAlchemy for model definitions and queries
-

Day 5: Authentication, Final Project, and Deployment

Module 10: Authentication & Security

- User Login with JWT (JSON Web Token)
- Password Hashing with bcrypt or passlib
- Protecting Endpoints with Tokens

Module 11: Capstone Project & Deployment

Capstone: Task Management App

- User Signup/Login

- Create and Manage Tasks
- Secure APIs with JWT
- Use PostgreSQL and SQLAlchemy

Labs:

- Full-stack backend development of Task Manager
- Host the project on GitHub and optionally deploy online