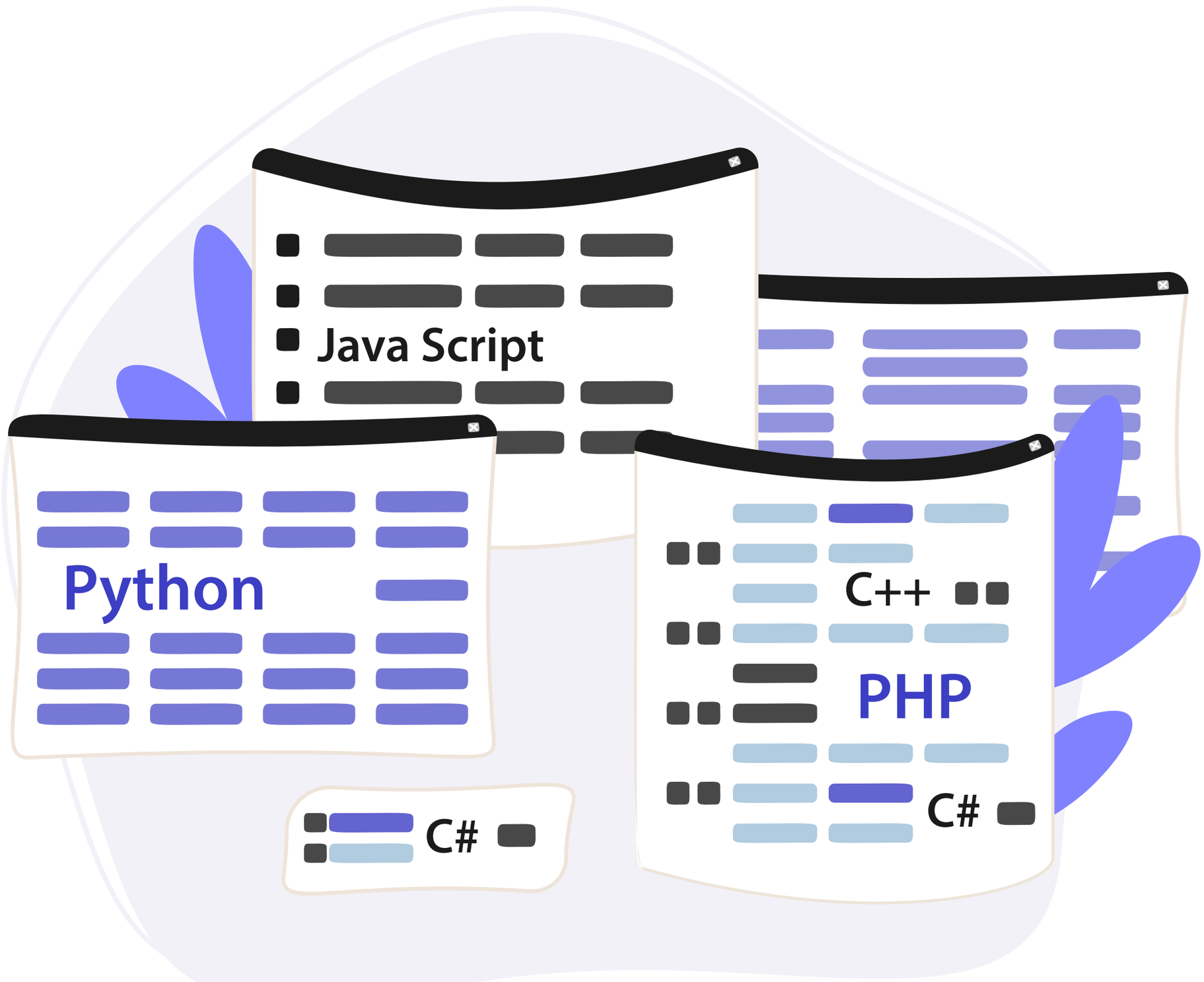
BOSSCODER ACADEMY

LEARN BACKEND

5 WEEKS TO PRO



Day 1-3

Understand How The Web Works

Topic to Cover

- Client-server architecture
- Basics of HTTP
- IP Addressing, DNS
- Request Response Cycle

Resources

1. How the Web Works
2. Basics of HTTP
3. Client Server Architecture
4. What is DNS? Domain Name System, DNS Server, and IP Address Concepts Explained

Questions

1. Imagine you're building a chat application. Is the chat server considered the client or the server in the client-server architecture? Explain your choice.

2. Perform a command in your terminal (e.g., nslookup) to retrieve the IP address of a specific domain name (e.g., www.example.com). Explain the output.

3. Describe the difference between HTTP and HTTPS.

4. Explain the difference between IPv4 and IPv6.

5. Open your browser's developer tools and inspect the network tab. Load a website and analyze the requests and responses exchanged during the page load.

6. In a client-server model, where does the processing of data primarily occur, on the client or the server?

Day 4-7

HTML and CSS

Topic to Cover

- Introduction to HTML tags and elements
- HTML document structure
- CSS syntax and selectors
- Styling elements and layouts
- CSS Block Model
- Responsive design principles (Media Queries)

Resources

1. https://www.freecodecamp.org/news/html-css-handbook-for-beginners/
2. https://www.w3schools.com/html/
3. https://www.w3schools.com/css/
4. https://developer.mozilla.org/en-US/docs/Learn/CSS
5. https://www.youtube.com/watch?v=G3e-cpL7ofc

Questions

1. Create an HTML structure for a simple blog post, including headings, paragraphs, images, and links.

2. Write a CSS selector to target all paragraphs within a specific class, and provide an example of a style you could apply to those paragraphs.

3. How can you style a specific HTML element using an inline style? Provide an example.

4. Create a CSS rule that sets the font color to red for all headings of a specific class.

5. Describe the purpose of media queries in responsive design. Write a media query that targets screens with a maximum width of 600px and adjusts the font size of a heading to 18px.

Day 8-11

Introduction to Back-end Frameworks - Node.js

Topic to Cover

- Setting up Node.js environment
- Modules in Node.js
- Asynchronous Concepts
- Event loops, Callbacks
- npm and Packages

Resources

1. https://nodejs.dev/en/learn/
2. https://www.youtube.com/watch?v=TIB_eWDSMt4
3. https://www.w3schools.com/nodejs/

Questions

1. Why is it important to have a package manager like npm when working with Node.js?

2. Differentiate between built-in modules and external modules in Node.js.

3. Create a JavaScript file with a function. Export the function using module.exports and import it in another file.

4. Explain the purpose of callbacks in Node.js and how they prevent blocking behavior.

5. Create a simple program that reads a file asynchronously using the fs module. Use a callback to handle the data.

6. How do you find, install, and manage packages using npm?

Day 12-14

Relational Databases

Topic to Cover

- Introduction to relational databases.
- Tables, rows, columns, and relationships.

Resources

1. https://www.oracle.com/in/database/what-is-a-relational-database/
2. https://cloud.google.com/learn/what-is-a-relational-database
3. System Design — SQL vs NoSQL. Concepts and considerations for SQL and... | by Larry | Peng Yang | Computer Science Fundamentals | Medium
4. How To Choose The Right Database?
5. System Design Interview Prep: SQL vs NoSQL Databases - Exponent
6. Databases: system design interview concepts (2 of 9)

Questions

1. Explain the concept of normalization in the context of relational databases. What are the benefits and drawbacks of normalization?

2. Compare the differences between INNER JOIN, LEFT JOIN, and RIGHT JOIN operations in SQL.

3. Explain the trade-offs between using a relational database and a NoSQL database for a specific use case.

4. Explain the concept of a self-join in SQL. Provide an example query that demonstrates a self-join.

5. Explain the concept of a primary key in a relational database. Why is it important, and how is it different from a unique key?

6. Given two tables, Customers and Orders, write an SQL query to retrieve all customers and their orders using an INNER JOIN.

Day 15-17

Non-Relational Databases (NoSQL)

Topic to Cover

- Introduction to NoSQL databases.
- Types of NoSQL databases: Document-based, Key-Value, Column-Family, and Graph.
- Schema flexibility in NoSQL databases.

Resources

1. What is NoSQL?
2. NoSQL Databases: An Overview
3. Introduction to NoSQL Databases
4. CAP Theorem Explained

Questions

1. Explain the key characteristics of NoSQL databases and how they differ from relational databases.
2. Discuss the concept of schema flexibility in NoSQL databases and why it's important. Provide an example.
3. Give an example of a real-world use case where a document-based NoSQL database like MongoDB would be a suitable choice.
4. What is eventual consistency in the context of NoSQL databases, and why is it important for distributed systems?
5. Explain the concept of sharding in NoSQL databases and how it helps in scalability. Provide a use case where sharding would be beneficial.
6. You are designing a backend for a content management system where users can create and edit articles with varying structures. Explain why you would choose MongoDB as the database system for this application and provide a basic schema example.

Day 18-21

Advanced Database Concepts

Topic to Cover

- ACID Properties
- Database sharding and partitioning
- Database indexing
- Concurrency Control

Resources

1. ACID Properties in DBMS - GeeksforGeeks
2. Database Sharding vs. Partitioning: What’s the Difference?
3. System Design — Sharding / Data Partitioning | by Larry | Peng Yang | Computer Science Fundamentals | Medium
4. An in-depth look at Database Indexing
5. Concurrency Control in DBMS - GeeksforGeeks

Questions

1. Explain the ACID properties in the context of database transactions. How do they ensure data integrity?

2. Discuss the trade-offs between ensuring strong ACID properties and achieving high performance in a database system.

3. Define database sharding and partitioning. How do they contribute to scalability?

4. What are the key factors to consider when selecting a sharding key for a database table?

5. Compare B-tree indexes and hash indexes. When would you choose one over the other?

6. What is a deadlock? Provide an example and describe strategies to prevent or resolve deadlocks.

7. In a high-traffic e-commerce website, discuss the trade-offs between ensuring strong ACID properties and achieving high performance in the database system.

Day 21-24

REST API

Topic to Cover

- HTTP Verbs
- Request and Response Format
- Statelessness
- Security and Authentication

Resources

1. https://www.restapitutorial.com/
2. https://www.youtube.com/watch?v=lsMQRaeKNDk
3. https://www.geeksforgeeks.org/rest-api-introduction/
4. https://rapidapi.com/learn/rest

Questions

1. Explain the purpose of HTTP verbs (GET, POST, PUT, DELETE) in RESTful APIs.

2. Create a simple Express.js server with routes for handling POST, PUT, and DELETE requests.

3. Explain the structure of an HTTP response, including status codes and response bodies.

4. Design an API response format for a user registration endpoint that returns a success message and a user ID.

5. Explain how statelessness contributes to scalability and simplicity in API design.

6. Explore API key authentication: Generate an API key, send it as a header in a request, and validate it on the server.

7. Explain the difference between authentication and authorization.

Day 25-29

Websockets and WebHooks

Topic to Cover

- Implementation of WebHooks
- Websocket protocols
- Use cases

Resources

1. https://hookdeck.com/webhooks/guides/when-to-use-webhooks#webhooks-or-pubsub
2. https://www.youtube.com/watch?v=6RvIKYgRFYQ
3. https://nonamesecurity.com/learn/api-vs-webhook-vs-websocket/

Questions

1. Describe a use case where webhooks are a suitable choice for communication.

2. How do webhooks typically handle data transmission between services?

3. How does real-time data exchange work in a WebSocket-based application?

4. Why might WebSockets be preferred over webhooks for real-time document collaboration?

Day 30-35

Websockets and WebHooks

Building a small project is an excellent way to learn backend development as it allows you to apply your knowledge in a practical context. Let's create a simple project idea and provide content to help you get started.

Project Idea: Task Manager API

Project Description

Create a basic task manager API that allows users to create, read, update, and delete tasks. Users should be able to register, log in, and manage their tasks. This project will cover fundamental backend development concepts such as routing, database interaction, authentication, and API endpoints.

Technologies to Use

For this project, you can use Node.js as the backend runtime and

Day 30-31

Database and User Authentication

- **Setup:** Create a new project directory and initialize it with Node.js and npm.
- **Server:** Set up a basic Express.js server with a simple "Hello World" route.

Database and User Authentication

- **Database:** Install and configure MongoDB for local development.
- **User Authentication:** Implement basic user registration and login functionality with password hashing.

Day 32-34

Task Management API

- **Task Model:** Define a schema for tasks and set up a MongoDB collection for tasks.
- **API Endpoints:** Create API endpoints for creating, reading, updating, and deleting tasks.
- **CRUD Operations:** Implement basic CRUD operations for tasks in your database.

User Tasks and Authorization

- **User-Specific Tasks:** Ensure that tasks are associated with the authenticated user.
- **Authorization Middleware:** Create middleware to protect routes that require authentication.

Day 35

Deployment

Deploy your project to a hosting platform like Heroku or AWS. Set up a production database and configure environment variables.

BOSSCODER ACADEMY

WHY BOSSCODER?

1000+ Alumni placed at Top Product-based companies.

More than **136% hike** for every **2 out of 3** working professional.

Average package of **24LPA.**

The syllabus is most **up-to-date** and the list of problems provided covers **all important topics.**

Lavanya
Meta

Course is very well **structured** and **streamlined** to crack any **MAANG** company

Rahul .
Google

EXPLORE MORE