ONE PAGE SUMMARY

IT & Software › Backend Development › Node.js

Introduction to Node.js

Unlock the power of asynchronous programming with Node.js!

DEFINITION

> Node.js is a JavaScript runtime built on Chrome's V8 engine, enabling server-side development. It allows developers to build scalable network applications using an event-driven, non-blocking I/O model.

KEY CONCEPTS

Event Loop

The event loop handles asynchronous operations, allowing Node.js to perform non-blocking I/O.

Non-blocking I/O

Node.js processes multiple requests simultaneously without waiting for any single operation to complete.

Modules

Node.js uses a modular architecture, allowing developers to include reusable code through modules.

NPM

Node Package Manager (NPM) is the default package manager for Node.js, facilitating easy installation of libraries.

REST APIs

Node.js is commonly used to build RESTful APIs, enabling communication between client and server.

EXAMPLES

→ Creating a simple web server with Node.js.

→ Using NPM to install Express framework.

→ Building a REST API for a to-do list application.

MEMORY TIPS

★ Remember 'Node' as the 'server' of JavaScript.

★ Think of NPM as the 'app store' for Node.js libraries.

★ Associate the event loop with 'events' happening in real-time.

COMMON MISTAKES

✗ Not understanding the asynchronous nature of Node.js.

✗ Neglecting error handling in callbacks.

✗ Confusing synchronous and asynchronous code execution.

QUICK RECAP

Node.js is a powerful tool for building server-side applications using JavaScript. Its event-driven architecture and non-blocking I/O model make it efficient for handling multiple requests simultaneously.