KOENIG
step forward

info@koenig-solutions.com

Node JS

Duration: 40 Hours (5 Days)

Overview

The Node.js course is designed to equip learners with the skills and knowledge necessary to build scalable and efficient web applications using Node.js, a powerful JavaScript runtime environment. It covers key concepts such as Asynchronous event-driven programming, Streaming data, File system operations, Using Express for web server development, and interacting with both SQL and NoSQL databases.Module 1: Understanding Node Environment sets the foundation by introducing the Node.js ecosystem, extending JavaScript capabilities to the server-side, understanding the V8 engine, and exploring the Global Process object.Module 2: Understanding Asynchronous event-driven programming dives into Node's Non-blocking I/O model, teaching learners about Event broadcasting, Event listeners, Timers, the Event loop, and Error handling with callbacks.The course then progresses to Streaming data, file system access, and web development with Express, providing hands-on experience with Routes, Middleware, and JSON handling. Integration with databases through CRUD operations is covered extensively, using both MySQL and MongoDB to give a rounded perspective on data persistence.Advanced topics include Connecting Node.js with Angular, allowing for the creation of full-stack applications. Learners will understand how to build Angular components, services, and handle events and forms.Finally, the course concludes with Deployment strategies, ensuring learners can launch their applications into a production environment. This comprehensive Node.js course is ideal for those looking to become proficient in modern back-end web development, and the practical knowledge gained will be invaluable for both personal and professional projects.

Audience Profile

Koenig Solutions' Node.js course offers comprehensive training in server-side development for aspiring and current web developers.

- Target audience for the Node.js course includes:
- Web Developers seeking to enhance their backend development skills.
- Frontend Developers looking to become Full Stack Developers.
- Software Engineers interested in building scalable network applications.
- IT Professionals wanting to learn asynchronous event-driven programming.
- Computer Science students or graduates aiming to specialize in JavaScript server-side development.
- Technical Leads and Managers needing to understand Node.js for project oversight.
- System Architects designing web application infrastructure.
- DevOps Engineers who integrate Node.js applications into their workflows.
- Quality Assurance Testers who require knowledge of Node.js for testing purposes.
- Entrepreneurs or Freelancers planning to build and deploy web applications independently.

Course Syllabus

Prerequisites: Knowledge of Java Script ES6 programming

Day 1: Introduction to Node.js and Basic Concepts

- What is Node.js?
- History and evolution of Node.js
- Understanding the Node.js runtime and event-driven architecture
- Installing Node.js and npm
- Using the Node.js REPL
- Understanding the project structure and package.json
- Understanding modules in Node.js
- Using built-in modules (fs, path, os)
- Creating and exporting custom modules
- Reading from and writing to files
- Working with directories
- Handling file system errors

Lab 1: Setting Up and Basic Modules

- Setting up a new Node.js project
- Creating and using custom modules
- Performing file operations using the fs module

Day 2: Asynchronous Programming and HTTP Module

- Callbacks in Node.js
- Introduction to Promises
- Using async/await for asynchronous code
- Understanding EventEmitter class
- Creating and handling custom events
- Working with readable and writable streams
- Using the http module
- Creating an HTTP server
- Handling basic routing and responses
- Parsing request data
- Handling POST requests
- Working with query strings and URL parameters

Lab 2: Asynchronous Code and HTTP Server

- Writing asynchronous functions with callbacks, Promises, and async/await
- Creating and using custom events
- Building a simple HTTP server and handling requests

Day 3: Express.js and Middleware

- Installing and setting up Express
- Basic routing with Express
- Serving static files
- Understanding middleware functions
- Creating custom middleware
- Using built-in middleware

- Handling route parameters
- Working with query strings
- Nested routes and routers
- Error handling middleware
- Debugging Express applications
- Using logging libraries (e.g., morgan)

Lab 3: Building an Express Application

- Setting up an Express project
- Implementing routes and middleware
- Handling errors and debugging issues

Day 4: Database Integration and Authentication

- Introduction to MongoDB
- Setting up and connecting to MongoDB using Mongoose
- Defining schemas and models
- Creating documents
- Reading documents with queries
- Updating and deleting documents
- Introduction to JSON Web Tokens (JWT)
- Setting up authentication routes
- Protecting routes with middleware
- Understanding sessions and cookies
- Using session middleware
- Implementing login and logout functionality

Lab 4: Database Operations and Authentication

- Setting up MongoDB and Mongoose
- Performing CRUD operations
- Implementing JWT-based authentication and session management

Day 5: Real-Time Applications and Deployment

- Introduction to WebSockets
- Setting up Socket.io with Node.js
- Creating a real-time chat application
- Designing RESTful routes
- Implementing CRUD operations for API endpoints
- Testing APIs with Postman
- Writing unit tests with Mocha and Chai
- Using Supertest for API testing
- Debugging techniques and tools
- Preparing a Node.js application for deployment
- Deploying to cloud platforms (Heroku, AWS)
- Best practices for Node.js development

Lab 5: Final Project and Deployment

- Building a real-time chat application
- Creating and testing a RESTful API
- Deploying the final project