

NODE.JS

COURSE CURRICULUM



ENROLL NOW



CALL US

+91 630 171 9270

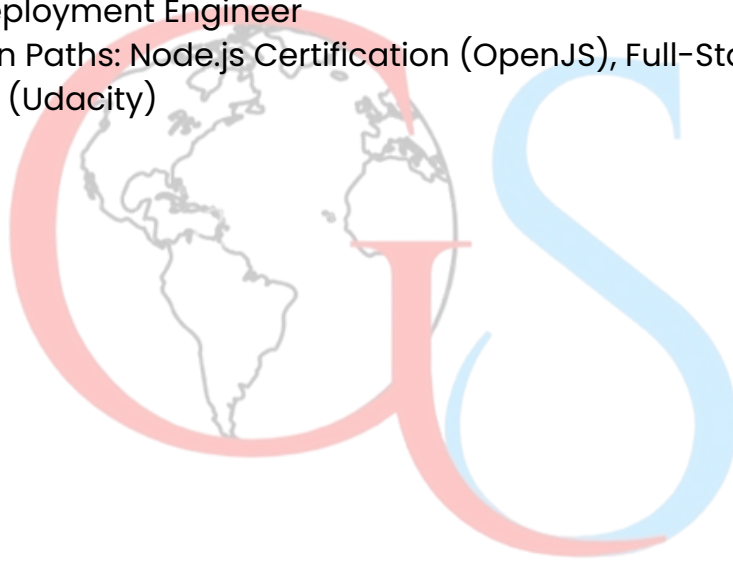
NODE.JS – COURSE CURRICULUM

ROAD MAP

- Learn how to build fast, scalable, and efficient back-end applications using Node.js
- Perfect for JavaScript Developers, Backend Engineers, and Full-Stack Developers
- Real-world use cases: APIs, Web Servers, Authentication, and Deployment

CAREER PATH

- Node.js Developer
- Backend Engineer
- Full-Stack Developer
- DevOps/Deployment Engineer
- Certification Paths: Node.js Certification (OpenJS), Full-Stack Developer Nanodegree (Udacity)



I N F O T E K H

COURSE CONTENT

Module 1: Getting Started with Node.js

- Introduction to Node.js
- What is Node.js and Why use it
- Installing Node.js
- Running JavaScript outside the browser

Module 2: Working with Modules & File System

- Using Core Modules
- Reading and Writing Files (Sync vs Async)
- Blocking vs Non-Blocking Operations
- Creating a Simple Web Server
- Routing and Building a Simple API

Module 3: HTML Templating & Custom Modules

- Building and Filling HTML Templates
- Parsing Variables from URLs
- Creating Custom Modules
- Introduction to NPM & package.json
- Types of Packages, Versioning, and Updates
- Using 3rd-Party Modules
- Setting Up Prettier in VS Code

Module 4: Backend Web Fundamentals

- HTTP in Action
- Front-End vs Back-End
- Static vs Dynamic vs API
- How Node.js Works – Node, V8, Threads

Module 5: Events, Streams & Asynchronous JavaScript

- Node.js Event Loop
- Event-Driven Architecture & Practice
- Introduction to Streams
- Requiring Modules in Depth
- Callbacks, Promises, and Async/Await
- Managing Async Functions & Multiple Promises

Module 6: Express.js Fundamentals

- Introduction to Express.js
- Basic Routing and RESTful API Design
- Handling GET, POST, PATCH, DELETE
- Middleware – Built-in, Custom, and 3rd-Party
- Creating and Mounting Multiple Routers
- Better File Structure and Param Middleware
- Serving Static Files
- Environment Variables & ESLint Setup

Module 7: MongoDB Integration

- Introduction to MongoDB
- Local & Hosted Setup (Atlas)
- CRUD Operations (Create, Read, Update, Delete)
- Using MongoDB Compass
- Connecting MongoDB to Node.js

Module 8: Error Handling in Express

- Debugging with ndb
- Unhandled Routes & Global Error Middleware
- 404 Not Found and Async Error Catching
- Handling Mongoose Errors (Validation, Duplicates, Invalid IDs)
- Managing Errors in Dev vs Production
- Handling Unhandled Rejections & Uncaught Exceptions

Module 9: Authentication, Authorization & Security

- User Modeling & Password Management
- JWT Authentication – Sign Up & Login
- Role-based Authorization
- Password Reset Flow using Nodemailer
- User Account Management (Update, Delete)
- Security Best Practices
- Cookies, Rate Limiting, HTTP Headers
- Data Sanitization & Preventing Parameter Pollution

Module 10: Git, Deployment & DevOps Basics

- Git & GitHub Setup
- Git Commands & Workflow
- Preparing App for Deployment
- Deploying to Heroku
- Enabling HTTPS, Handling SIGTERM
- Implementing CORS

LEARNING OUTCOMES

- Build real-time and RESTful APIs using Node.js and Express
- Integrate MongoDB for full-stack development
- Master error handling and secure authentication
- Deploy full apps on cloud platforms like Heroku
- Handle production-level security and performance issues



Thank you



Thank You for Going Through Core Node.js Curriculum
We hope this guide has provided a clear and structured learning path to strengthen your skills in Node.js.

📌 **NEXT STEPS**

- Start practicing with real-world use cases and hands-on exercises
- Build personal or client-based projects for your portfolio
- Keep exploring updates and best practices in the industry
- Join discussions and stay connected with the community

📞 **Need Help or Guidance?**

Feel free to contact our course support team:
Course Coordinator

GS Infotekh

✉ **contact@gsinfotekh.com**

🌐 **www.gsinfotekh.com**

☎ **+91 630 171 9270**