

Node Js

1. Introduction to Node.js

- What is Node.js?
- Features of Node.js
- Node.js vs traditional server-side technologies (e.g., PHP, Java)
- Installing Node.js and npm
- Running your first Node.js script

2. Node.js Basics

- Understanding `console.log()`, `require()`, `module.exports`
- Global objects: `__dirname`, `__filename`, `process`, `setTimeout()`
- Creating and running basic modules
- Node.js REPL (Read-Eval-Print Loop)

3. Node Package Manager (npm)

- Installing packages
- Local vs global packages
- Semantic versioning
- Creating and managing `package.json`
- Useful packages for beginners (e.g., `chalk`, `nodemon`)

4. File System Module (`fs`)

- Reading from files
- Writing to files
- Appending and deleting files
- Working with file paths (`path` module)

5. HTTP Module

Node Js

- Creating a basic HTTP server
- Handling requests and responses
- Setting headers and status codes
- Serving static files

6. Express.js (Minimal Intro)

- Installing and using Express
- Creating a simple web server
- Defining routes (GET, POST)
- Handling query parameters and request bodies
- Serving static files with Express

7. Middleware (Basics)

- What is middleware?
- Creating custom middleware
- Using built-in middleware (`express.json()`, `express.static()`)

8. Asynchronous Programming

- Callback functions
- Promises and `.then()`
- `async/await`
- Error handling in async functions

9. Working with APIs

- Creating a simple REST API with Express
- CRUD operations (Create, Read, Update, Delete)
- Handling JSON data

Node Js

10. Basic Database Connectivity (Optional)

- Introduction to MongoDB or any database
- Using mongoose (MongoDB ODM) or mysql package
- Connecting to database and performing simple queries

11. Error Handling & Debugging

- Try/catch blocks
- `next()` in Express for error handling
- Using nodemon and logging for debugging