

## 15-Day React.js and Node.js Course

### *Day 1-2: Introduction to JavaScript and Node.js*

- JavaScript fundamentals (ES6+ features)
- Node.js installation and setup
- Understanding the Node.js runtime and event-driven architecture
- Basic server creation with Node.js and Express.js
- Handling HTTP requests (GET, POST, PUT, DELETE)

### *Day 3-4: Node.js and Express.js Basics*

- Setting up Express.js to handle routes and requests
- Middleware in Express.js
- Handling JSON and form data in Express
- Introduction to RESTful API design

### *Day 5-7: Introduction to React.js*

- Setting up a React environment (Create React App)
- Understanding components, props, and state
- Functional components and React hooks (useState, useEffect)
- Handling events and form submissions in React

### *Day 8-10: Connecting React with Node.js Backend*

- Setting up Axios to make HTTP requests from React
- Fetching data from Node.js backend and displaying it in React
- Managing state in React with hooks
- Building dynamic forms in React

### *Day 11-13: Authentication and Authorization*

- Implementing user authentication with JWT (JSON Web Tokens)
- Storing and managing tokens in localStorage or cookies
- Protecting routes and ensuring only authorized users can access certain pages
- Basic session management with JWT

### *Day 14-15: Final Project and Review*

- Build a simple full-stack application with React and Node.js
- Testing, debugging, and deployment on platforms like Heroku

- Final project review and presentation

## **30-Day React.js and Node.js Course**

### ***Week 1: JavaScript, Node.js, and Express.js Basics***

- JavaScript fundamentals (ES6 features, asynchronous programming)
- Node.js fundamentals and Express.js setup
- Building RESTful APIs with Node.js and Express.js
- Handling HTTP requests (GET, POST, PUT, DELETE)
- Introduction to MongoDB and Mongoose for data storage

### ***Week 2: Advanced Node.js and Express.js***

- Middleware functions in Express.js
- Handling form data and JSON in Node.js
- Integrating MongoDB and Express using Mongoose
- Error handling in Express applications
- Validating and sanitizing user input

### ***Week 3: Introduction to React.js and Frontend Development***

- React component architecture (function components, class components)
- React hooks (useState, useEffect)
- React Router for routing and navigation
- State management in React (props, useState, useContext)

### ***Week 4: MERN Stack Integration and Authentication***

- Connecting React with Node.js and MongoDB using Axios
- Implementing JWT-based authentication and user login/registration
- Creating secure routes and protecting sensitive data
- Testing and debugging full-stack applications
- Deployment (Heroku, Netlify, or DigitalOcean)

### ***Day 29-30: Final Project and Review***

- Build a full-stack application with React.js and Node.js
- Handling authentication, routing, and form submissions
- Final project testing, debugging, and deployment
- Project review and feedback

## 45-Day React.js and Node.js Course

### *Week 1-2: JavaScript, Node.js, and Express.js Basics*

- Advanced JavaScript concepts (closures, promises, async/await)
- Setting up Node.js and Express.js server
- Building APIs and handling routing in Express
- Integrating MongoDB with Node.js using Mongoose
- Introduction to API testing with Postman

### *Week 3: Advanced React.js*

- React component lifecycle and class components
- React hooks (useEffect, useRef, useReducer)
- Handling forms and validations in React
- React Router for navigation and dynamic routing
- State management in React (Context API)

### *Week 4: Backend Development with Node.js*

- Authentication using JWT (JSON Web Tokens)
- Working with RESTful APIs in Node.js
- Integrating third-party services in Node.js applications (APIs, file storage)
- Using environment variables for configuration
- Error handling and logging in Node.js

### *Week 5: Full-Stack Development with React.js and Node.js*

- Full-stack integration between React and Node.js
- Fetching data from backend and displaying it in React components
- Protecting backend routes and managing user sessions with JWT
- Managing application state in React and Node.js
- File handling and image uploads in Node.js

### *Day 44-45: Final Project and Review*

- Build a complete full-stack MERN application
- Handle authentication, routing, and CRUD operations
- Project testing, debugging, and optimization
- Final review, presentation, and deployment

## 60-Day React.js and Node.js Course

### ***Month 1: JavaScript, Node.js, and Express.js Mastery***

- Advanced JavaScript features (destructuring, spread/rest operators, async/await)
- Node.js fundamentals, modules, and file system management
- Building REST APIs with Express.js and connecting to MongoDB
- Security practices: Hashing passwords with bcrypt, input validation, and error handling
- Integrating Node.js with third-party APIs (email, payment gateways)

### ***Month 2: Advanced React.js and MERN Stack Integration***

- Understanding React component lifecycle and state management
- Using hooks in React (useEffect, useReducer)
- Integrating React with Express API using Axios
- Implementing authentication and authorization with JWT
- Handling file uploads and static assets in Express
- Creating full-stack applications with React.js and Node.js

### ***Day 59-60: Final Project and Review***

- Build a complete MERN stack application with authentication, state management, and API integration
- Testing and debugging, implementing error handling
- Deployment and performance optimization
- Final project review and presentation

## **90-Day React.js and Node.js Course**

### ***Month 1: Advanced JavaScript, Node.js, and Express.js***

- Deep dive into JavaScript concepts (closures, promises, async/await, ES6 features)
- Node.js core modules and building scalable applications
- Working with MongoDB using Mongoose (queries, relationships)
- Building RESTful APIs and integrating third-party services
- Authentication and session management (JWT, Passport.js)

### ***Month 2: Advanced React.js and MERN Stack***

- Advanced React concepts (state management, Context API, hooks)
- Integrating React with Node.js backend using Axios

- Managing application state with Redux (optional)
- Handling authentication with JWT and Passport.js
- Implementing file uploads, notifications, and real-time features

### ***Month 3: Full-Stack Development, Optimization, and Deployment***

- Optimizing React and Node.js applications for performance
- Real-time applications with WebSockets (Socket.io)
- Deploying applications to cloud platforms (AWS, Heroku, DigitalOcean)
- Scaling and optimizing MongoDB queries
- Implementing unit and integration testing for React and Node.js
- CI/CD pipeline setup

### ***Day 89-90: Final Project and Review***

- Build a large-scale application using MERN stack (user authentication, file uploads, API integration)
- Testing, debugging, performance tuning, and deployment
- Final project presentation and deployment review

## **180-Day React.js and Node.js Course**

### ***Month 1-2: JavaScript, Node.js, and Express.js Mastery***

- Advanced JavaScript (asynchronous programming, design patterns, ES6+)
- Node.js architecture and modules
- Building RESTful APIs with Express.js
- Connecting Node.js with databases (MongoDB, MySQL)
- Integrating third-party APIs and services

### ***Month 3: Advanced React.js Concepts***

- React hooks and Context API for state management
- React Router for advanced routing
- Performance optimization in React applications
- Real-time features (WebSockets, push notifications)
- Advanced form handling and validation in React

### ***Month 4: Full-Stack MERN Development***

- Full-stack app development with React and Node.js

- Integrating RESTful APIs, JWT authentication, and third-party services
- Creating real-time applications (chat, notifications, collaborative features)
- Setting up user roles and permissions
- Integrating payment gateways, email, and social media APIs

### ***Month 5: Optimization, Deployment, and Security***

- Optimizing React and Node.js apps for performance
- Securing applications (input validation, encryption, HTTPS)
- Handling scalability issues, and setting up caching
- Deploying applications to cloud platforms (AWS, DigitalOcean)
- Setting up continuous integration and deployment (CI/CD)

### ***Month 6: Final Project and Professional Development***

- Build a large-scale application with full authentication, API integration, and performance optimization
- Implement advanced features like file handling, real-time collaboration, and notifications
- Final project testing, debugging, and deployment strategies
- Professional development: Git workflow, project management, and teamwork
- Final project review, feedback, and career advice