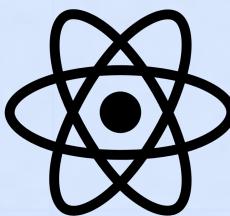




45 DAY CHALLENGE

To Learn

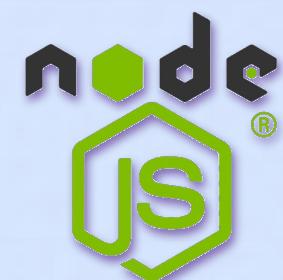
MERN Stack



ex



MERN
Stack





Disclaimer

Everyone learns differently.

A structured approach empowers you to tackle
new challenges effectively.

This Doc will guide you through the process.

Introduction and Setup

Day 1: Introduction to JavaScript

- JavaScript fundamentals: variables, data types, functions, loops.

Resources

Day 2: HTML and CSS Basics

- Learn HTML tags, CSS styling, and responsive design.

Resources 1 | Resources 2

Day 3: JavaScript DOM Manipulation

- Manipulating DOM elements using JavaScript.

Resources

Day 4: Introduction to Node.js

- Setting up Node.js, npm basics.

Resources

Day 5: Setting up MongoDB

- Installing MongoDB, basic CRUD operations.

Resources

Day 6: Review and Practice

- Review JavaScript, HTML, CSS, and Node.js basics.
 - Build a simple static website.
-

Day 7: Weekly Project

- Create a small project integrating HTML, CSS, JavaScript, and Node.js.

Weekly Checklist

- Understand JavaScript fundamentals.
- Get comfortable with HTML and CSS basics.
- Learn basic Node.js and npm setup.
- Perform CRUD operations with MongoDB.

Frontend Development with React

Day 8: Introduction to React

- Basics of React, JSX, components, props.

[Resources](#)

Day 9: React State Management

- Understanding state and lifecycle methods.

[Resources](#)

Day 10: React Hooks

- Using hooks like useState and useEffect..

[Resources](#)

Day 11: React Router

- Implementing routing in React applications.

[Resources](#)

Day 12: Forms in React

- Handling forms and input in React.

[Resources](#)

Day 13: State Management with Redux

- Introduction to Redux, setting up a simple store.

[Resources](#)

Day 14: Weekly Project

- Build a small React project (e.g., Todo App).

[Resources](#)

Weekly Checklist

- Understand React basics and JSX.
- Learn state management and lifecycle methods.
- Implement routing and forms in React.
- Manage state with Redux.

Backend Development with Node.js and Express.js

Day 15: Introduction to Express.js

- Setting up Express, basic routing.

Resources

Day 16: Middleware in Express.js

- Understanding and using middleware.

Resources

Day 17: RESTful APIs

- Building RESTful APIs with Express.

Resources

Day 18: Authentication and JWT

- Implementing authentication using JWT.

[Resources](#)

Day 19: MongoDB Integration

- Connecting MongoDB with Express, CRUD operations.

[Resources](#)

Day 20: Mongoose ODM

- Using Mongoose for data modeling.

[Resources](#)

Day 21: Weekly Project

- Build a small REST API with Express and MongoDB.

[Resources](#)

Weekly Checklist

- Understand Express.js basics and routing.
- Implement middleware and build RESTful APIs.
- Integrate MongoDB with Express.
- Use Mongoose for data modeling and CRUD operations.

Database Management with MongoDB

Day 22: MongoDB Basics Review

- Review MongoDB basics, CRUD operations.

Resources

Day 23: Mongoose Schemas and Models

- Deep dive into Mongoose schemas and models.

Resources

Day 24: Aggregation and Indexing

- Using aggregation framework and indexes in MongoDB.

Resources

Day 25: Data Validation

- Implementing data validation in Mongoose.

Resources

Day 26: Pagination and Sorting

- Implementing pagination and sorting.

Resources

Day 27: Authentication and Authorization

- Advanced authentication and role-based access control.

Resources

Day 28: Weekly Project

- Build a CRUD application with advanced MongoDB features.

Resources

Weekly Checklist

- Review and perform CRUD operations in MongoDB.
- Understand and implement Mongoose schemas and models.
- Use MongoDB aggregation and indexing.
- Implement data validation and pagination.

Full Stack Integration and Deployment (Part 1)

Day 29: Setting Up MERN Project

- Review MongoDB basics, CRUD operations.

Resources

Day 30: Connecting Frontend and Backend

- Integrating React frontend with Express backend.

Resources

Day 31: Building a Full CRUD Application

- Develop a full CRUD application (e.g., blog, eCommerce).

Resources

Day 32: Testing with Jest and Mocha

- Writing tests for your application.

[Resources 1](#) | [Resources 2](#)

Day 33: CI/CD with Jenkins and GitHub Actions

- Setting up continuous integration and deployment.

[Resources 1](#) | [Resources 2](#)

Day 34: Error Tracking with Sentry

- Implementing error tracking.

[Resources](#)

Day 35: Performance Optimization

- Optimizing the performance of your application.

[Resources](#)

Weekly Checklist

- Set up and structure a MERN project.
- Integrate frontend and backend.
- Develop a full CRUD application.
- Perform testing and deploy the application.

Full Stack Integration and Deployment (Part 2)

Day 36: Security Best Practices

- Implementing security best practices.

Resources

Day 37: Deployment to Heroku and Vercel

- Deploying your application to Heroku and Vercel.

Resources 1 | Resources 2

Day 38: Final Project Review and Launch

- Final review of your project, testing, and launch.

Resources

Day 39: Review and Practice

- Review all concepts learned and practice.
-

Day 40: Weekly Project

- Create a complete MERN stack project integrating all components.

Weekly Checklist

- Implement security best practices.
- Deploy applications to Heroku and Vercel.
- Review and test the final project.

Advanced Topics and Final Projects

Day 41: Advanced React Concepts

- Context API, advanced hooks, performance optimization.

[Resources 1](#) | [Resources 2](#)

Day 42: Advanced Node.js Concepts

- Streams, buffers, and advanced file handling.

[Resources](#)

Day 43: GraphQL with Node.js

- Introduction to GraphQL and integrating with Node.js.

[Resources](#)

Day 44: WebSockets with Socket.io

- Real-time communication using WebSockets.

Resources

Day 45: Final Review and Capstone Project

- Review all concepts and build a capstone project.

Resources

Weekly Checklist

- Understand advanced React and Node.js concepts.
- Integrate GraphQL with Node.js.
- Implement real-time communication with WebSockets.
- Complete a final capstone project.