



Frontend Development – OA25FD01

Beginner to Expert Course

Course Description

Master the art of building responsive, interactive, and visually appealing user interfaces with this comprehensive frontend development course. From HTML and CSS basics to advanced React and TypeScript, this course covers everything you need to create modern web applications. Designed for all skill levels, it's compatible with tools and frameworks used in 2025.

What You'll Learn

- Build responsive websites with HTML, CSS, and JavaScript
- Master modern frontend frameworks like React and Next.js
- Create accessible and performant user interfaces
- Work with APIs and dynamic data
- Deploy frontend applications to the cloud

Course Requirements

- No prior programming knowledge required
- A computer with a modern web browser and code editor (e.g., VS Code)

Who This Course Is For

- Beginners starting their journey in frontend development
- Developers looking to specialize in user interface design
- Professionals aiming to build modern, responsive web applications

Syllabus (35 Sessions)

1. **Web Development Fundamentals**
 - Introduction to frontend development
 - Setting up VS Code and browser dev tools
 - HTML basics: Tags, attributes, semantic elements
 - Common coding shortcuts
2. **Styling with CSS**
 - CSS syntax, selectors, and properties
 - Colors, fonts, and the CSS Box Model
 - Introduction to Flexbox
 - Using CSS variables
3. **JavaScript Basics**
 - Variables, data types, and operators



- Functions and scope
- DOM manipulation
- Event handling (click, input events)

4. Responsive Design

- Mobile-first design with media queries
- CSS Grid for layouts
- Handling images and media
- Testing across devices

5. Advanced CSS

- Animations and transitions
- CSS preprocessors (Sass basics)
- Using Tailwind CSS
- Accessibility with ARIA roles

6. Intermediate JavaScript

- Arrays, objects, and loops
- ES6+: Arrow functions, destructuring, modules
- Event delegation
- Debugging with browser tools

7. Version Control with Git

- Setting up Git and GitHub
- Basic Git commands: Commit, push, pull
- Branching and merging
- Collaborating on GitHub

8. Working with APIs

- Understanding REST APIs
- Using fetch API for data
- Handling JSON and errors
- Displaying API data in UI

9. Practice Session

- Build a static portfolio website
- Congratulations! You've built a frontend foundation.

10. Introduction to React

- Setting up React with Vite
- JSX and components
- Props and state
- Styling with Tailwind CSS

11. React Hooks

- useState and useEffect
- Building reusable components
- Handling forms and validation
- Fetching data in React

12. React Routing

- Setting up React Router
- Creating multi-page applications
- Dynamic routing



- Nested routes

13. State Management in React

- Using Context API
- Introduction to Redux
- Managing complex state
- Persisting state with local storage

14. Advanced React Patterns

- Custom Hooks
- Compound components
- Render props
- Error boundaries

15. Introduction to TypeScript

- TypeScript setup and benefits
- Types and interfaces
- Using TypeScript with React
- Type-safe components

16. Server-Side Rendering

- Introduction to Next.js
- Static site generation
- Server-side rendering
- API routes in Next.js

17. Practice Session

- Build a blog application with React
- Congratulations! You're an intermediate frontend developer.

18. Advanced CSS Frameworks

- Deep dive into Tailwind CSS
- Building custom themes
- Optimizing CSS for performance
- CSS-in-JS (e.g., styled-components)

19. Frontend Testing

- Unit testing with Jest
- Component testing with React Testing Library
- Snapshot testing
- Mocking APIs

20. Performance Optimization

- Lazy loading components
- Code splitting in React
- Optimizing images and assets
- Measuring performance with Lighthouse

21. Accessibility Deep Dive

- WCAG guidelines
- Semantic HTML and ARIA
- Keyboard navigation
- Testing for accessibility

22. Animations in Frontend



- CSS animations and keyframes
- Using Framer Motion in React
- Creating smooth transitions
- Optimizing animations for performance

23. Working with Web APIs

- Web Storage API
- Geolocation API
- Drag-and-drop API
- Building interactive features

24. Deploying Frontend Apps

- Deploying to Vercel
- Configuring environment variables
- Setting up CI/CD with GitHub Actions
- Domain setup and HTTPS

25. Practice Session

- Build and deploy a task manager UI
- Congratulations! You've mastered frontend development.

26. Project 1: Portfolio Website

- Create a multi-page portfolio with React
- Add animations with Framer Motion
- Style with Tailwind CSS
- Deploy to Vercel

27. Project 2: E-Commerce UI

- Build product listing and cart pages
- Implement search and filters
- Use Redux for state management
- Style with Tailwind CSS

28. Project 3: Real-Time Dashboard

- Create a dashboard with React
- Integrate Chart.js for data visualization
- Fetch real-time data via WebSockets
- Deploy to Vercel

29. Project 4: Blog with Next.js

- Build a blog with Next.js
- Implement static site generation
- Add a markdown-based CMS
- Deploy with CI/CD

30. Project 5: Social Media UI

- Create a social media feed UI
- Implement infinite scroll
- Add like and comment features
- Test with React Testing Library

31. Advanced State Management

- Redux Toolkit for efficient state
- Zustand for lightweight state management



- Managing asynchronous data
- Persisting state across sessions

32. Progressive Web Apps (PWAs)

- Introduction to PWAs
- Service workers for offline support
- Caching strategies
- Testing PWA functionality

33. Project 6: Weather App

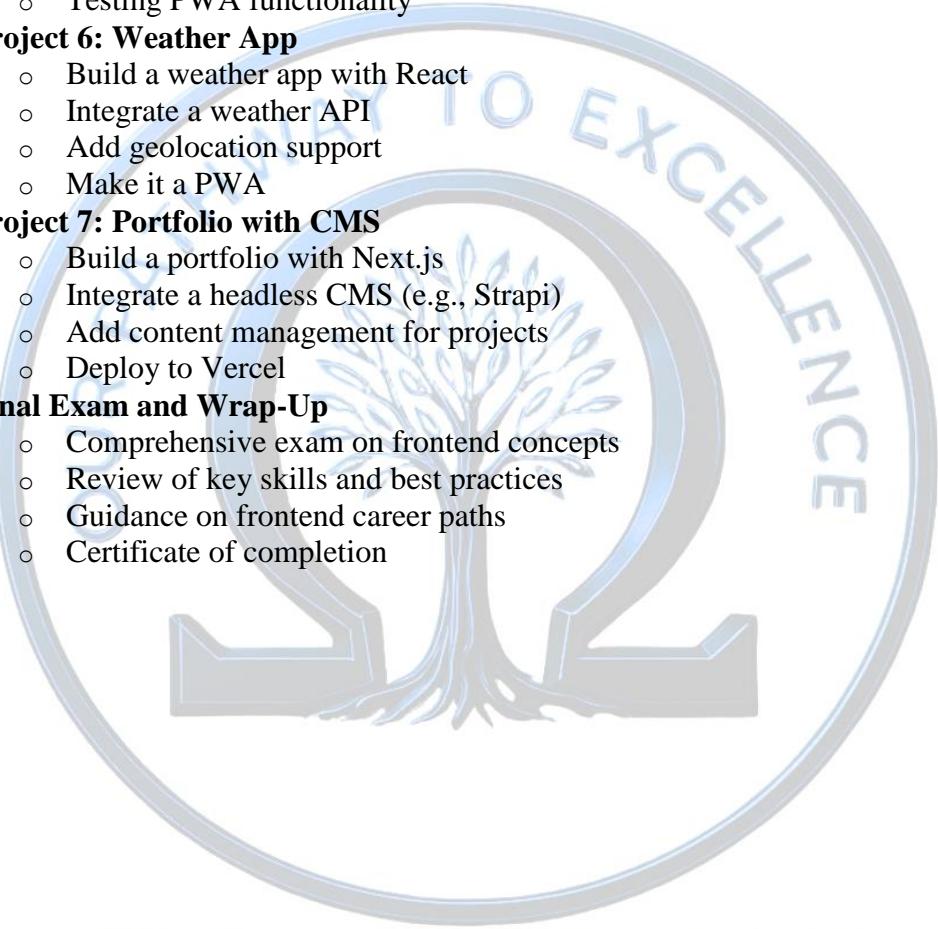
- Build a weather app with React
- Integrate a weather API
- Add geolocation support
- Make it a PWA

34. Project 7: Portfolio with CMS

- Build a portfolio with Next.js
- Integrate a headless CMS (e.g., Strapi)
- Add content management for projects
- Deploy to Vercel

35. Final Exam and Wrap-Up

- Comprehensive exam on frontend concepts
- Review of key skills and best practices
- Guidance on frontend career paths
- Certificate of completion



OMEGA'S
ACADEMY