



# ELEVO LABS

LEARN. INNOVATE. ELEVATE.

## Course Overview

---

### Course Details

**Course Name:** React - The Complete Guide: From beginner to Mastery

**Instructor:** Alex Ray

**Created On:** 16 June, 2025

**Updated On:** 16 June, 2025

**Price:** 10000 INR

**Duration:** 7 Weeks

**Modules:** 19

**Language:** English, Hindi

**Level:** Beginner

**Certifications:** Yes

## Course Description

---

This is the definitive guide to mastering React. We've structured this course to be the only one you'll ever need, taking you from the absolute fundamentals of web development to building complex, production-ready React applications. We leave no stone unturned.

The journey begins with a refresher on the essential modern JavaScript features that are critical for React development. We then dive into React's core philosophy—building applications with components, JSX, state, and props. You'll gain a deep, practical understanding of React Hooks, mastering everything from `useState` and `useEffect` to creating your own custom hooks for reusable logic.

As you advance, you'll tackle enterprise-level challenges. We dedicate multiple modules to advanced state management, covering the Context API in-depth before mastering Redux Toolkit for large-scale applications. You'll learn advanced component patterns, performance optimization techniques, and how to write professional, testable code. The course also includes dedicated modules on integrating with TypeScript and exploring the wider React ecosystem, including the Next.js framework.

## What You'll Learn:

- Core JavaScript concepts essential for React development.
- Deep understanding of React's component model, JSX, state, and props.
- Mastery of all React Hooks, including creating custom hooks.
- Advanced component design patterns (HOCs, Render Props).
- In-depth state management with Context API and Redux Toolkit.
- Robust client-side routing with React Router.
- Integrating TypeScript for building scalable and type-safe applications.
- Professional testing workflows with Jest and React Testing Library.
- Performance optimization, code-splitting, and profiling.
- Building and deploying full-stack applications with frameworks like Next.js.

## Who Is This Course For?

This course is designed for anyone committed to becoming a professional React developer. It's a perfect fit for absolute beginners who want a clear path, intermediate developers seeking to fill knowledge gaps and learn advanced patterns, and experienced developers who want to master the modern React ecosystem.

## Prerequisites:

- **Solid JavaScript Fundamentals:** A strong understanding of modern JavaScript (ES6+ features like `let`/const``, arrow functions, Promises, Async/Await, array methods, destructuring).
- **Basic HTML & CSS:** Familiarity with HTML structure and basic CSS styling.
- **Command Line Basics:** Comfort with basic terminal commands (e.g., `cd``, `npm install``).
- **A Computer/Laptop:** A working computer (Windows, macOS, or Linux) with Node.js and NPM installed.
- **A Willingness to Learn:** Enthusiasm for building dynamic user interfaces.

Includes numerous mini-projects, a capstone full-stack project, and a certificate upon completion. Available in English and Hindi.

## Course Curriculum Details

---

### Module 1: Module 1: Foundations & The 'Why' of React

4 Lessons

1. Course Introduction & What is React? (Declarative vs. Imperative) **15min**
2. Setting Up a Professional Development Environment (Vite, VSCode, Node.js) **25min**
3. Understanding React's Core Philosophy: Component-Based Architecture **20min**
4. Recap & Practical Task: Setup Your First React Project **30min**

### Module 2: Module 2: Essential JavaScript for React

6 Lessons

5. ES6+ Refresher: `let`/`const`, Arrow Functions, Template Literals **20min**
6. Mastering Array Methods: `map`, `filter`, `reduce` for Data Transformation **25min**
7. Destructuring and the Spread/Rest Operator (Objects & Arrays) **20min**
8. Understanding Asynchronous JavaScript: Promises & Async/Await **25min**
9. Working with JavaScript Classes (Brief Overview for Component Understanding) **15min**
10. Recap & Practical Task: Practice Advanced JavaScript Concepts with React-like Data **40min**

## Module 3: Module 3: React's Core Concepts (Components & JSX)

5 Lessons

1. 11. Creating Your First Functional Components **20min**
2. 12. Understanding and Using JSX (JavaScript XML) In-Depth **30min**
3. 13. Component Composition: Building UIs from Smaller Parts **30min**
4. 14. Passing Data with Props (Properties) and PropTypes **30min**
5. 15. Recap & Practical Task: Build a Static Component Tree **45min**

## Module 4: Module 4: State, Events, and Rendering Dynamic UIs

5 Lessons

1. 16. Managing Component State with the `useState` Hook **30min**
2. 17. Handling User Events (Click, Change, Submit) **25min**
3. 18. Conditional Rendering: Displaying Content Conditionally **25min**
4. 19. Rendering Lists of Data (`map()` and `key` Prop) **25min**
5. 20. Recap & Practical Task: Create an Interactive Counter or To-Do List **50min**

## Module 5: Module 5: Styling React Applications

4 Lessons

1. 21. Traditional CSS Strategies: Global Styles, CSS Modules, and Inline CSS **25min**
2. 22. CSS-in-JS: Styling with Styled-Components **30min**

3. 23. Utility-First Styling with Tailwind CSS (Integration & Basic Usage) **30min**

4. 24. Recap & Practical Task: Style a Component Using Different CSS Approaches  
**50min**

## Module 6: Module 6: Managing Side Effects with `useEffect`

**5 Lessons**

1. 25. Introduction to Side Effects in Functional Components **20min**

2. 26. Fetching Data from APIs with `useEffect` (GET Requests) **30min**

3. 27. Mastering the Dependency Array: When to Re-run Effects **35min**

4. 28. Cleanup Functions: Preventing Memory Leaks and Unwanted Behavior **30min**

5. 29. Recap & Practical Task: Fetch and Display Data from a Public API **1hr**

## Module 7: Module 7: Working with Forms

**5 Lessons**

1. 30. Controlled Components for Form Handling **30min**

2. 31. Uncontrolled Components and the `useRef` Hook for Forms **25min**

3. 32. Basic Form Validation and Handling User Input Errors **30min**

4. 33. Using Popular Form Libraries like React Hook Form **35min**

5. 34. Recap & Practical Task: Build a User Registration Form with Validation  
**1hr 15min**

## Module 8: Module 8: Advanced Hooks & Custom Hooks

**6 Lessons**

1. 35. Managing Complex State with `useReducer` **35min**

2. 36. Performance Optimization Hooks: `useMemo` for Memoizing Values **35min**

3. 37. Performance Optimization Hooks: `useCallback` for Memoizing Functions **35min**

4. 38. Accessing the DOM with `useRef` (Beyond Forms) **20min**

5. 39. Building Reusable Logic with Custom Hooks **30min**

6. 40. Recap & Practical Task: Create a Custom Data Fetching Hook **1hr**

## Module 9: Module 9: State Management with the Context API

5 Lessons

1. 41. Understanding Prop Drilling and Its Challenges **15min**

2. 42. Sharing State Globally with `createContext` and `useContext` **35min**

3. 43. Structuring Context Providers for Scalability and Performance **25min**

4. 44. Combining Context with `useReducer` for Complex State **30min**

5. 45. Recap & Practical Task: Implement Global Theme Switching with Context **1hr**

## Module 10: Module 10: Client-Side Routing with React Router

7 Lessons

1. 46. Setting up Client-Side Routing with React Router DOM **25min**

2. 47. Defining Routes, `Link` Components, and `BrowserRouter` **30min**

3. 48. Dynamic Routes and URL Parameters (`useParams` Hook) **30min**

4. 49. Nested Routes and Outlet Component **35min**

5. 50. Programmatic Navigation (`useNavigate` Hook) **25min**

6. 51. Protected Routes and Authentication Flow **30min**

7. 52. Recap & Practical Task: Build a Multi-Page Application with Routing **1hr 15min**

## Module 11: Module 11: Global State Management with Redux Toolkit

**8 Lessons**

1. 53. Why Redux for Large-Scale Applications? (Core Principles Revisited) **20min**

2. 54. Introduction to Redux Toolkit: Simplifying Redux Development **30min**

3. 55. Configuring the Redux Store with `configureStore` **30min**

4. 56. Defining Reducers and Actions with `createSlice` **40min**

5. 57. Connecting Redux Store to React Components (`useSelector`, `useDispatch`) **35min**

6. 58. Asynchronous Logic and Data Fetching with `createAsyncThunk` **40min**

7. 59. Using Redux DevTools Extension for Debugging **15min**

8. 60. Recap & Practical Task: Implement Global State for a Shopping Cart with Redux Toolkit **1hr 30min**

## Module 12: Module 12: Advanced Component Patterns

**6 Lessons**

1. 61. Higher-Order Components (HOCs): Reusable Logic **30min**

2. 62. Render Props Pattern for Flexible Component Logic **30min**

3. 63. Comparing HOCs and Render Props with Modern Hooks **20min**

4. 64. Compound Components Pattern **25min**

5. 65. Portals: Rendering Content Outside Component Hierarchy **20min**

6. 66. Recap & Practical Task: Build a Reusable Modal Component using Portals/Render Props **1hr**

## Module 13: Module 13: React Performance Optimization

**6 Lessons**

1. 67. Identifying Performance Bottlenecks with React DevTools Profiler **25min**

2. 68. Memoizing Components with `React.memo` (PureComponent equivalent) **25min**

3. 69. Code-Splitting with `React.lazy` and `Suspense` for Faster Loads **30min**

4. 70. Virtualization for Large Lists (Conceptual) **20min**

5. 71. Optimizing Context API Usage for Performance **20min**

6. 72. Recap & Practical Task: Optimize a Performance-Heavy Component **50min**

## Module 14: Module 14: Testing React Applications

**6 Lessons**

1. 73. Testing Philosophy & Setup (Jest & React Testing Library) **30min**

2. 74. Writing Unit Tests for Components and Hooks (Snapshot Testing, Basic Queries) **40min**

3. 75. Simulating Events and Testing User Interactions **35min**

4. 76. Mocking API Calls and External Dependencies **30min**

5. 77. Test Driven Development (TDD) Approach (Conceptual) **20min**

6. 78. Recap & Practical Task: Write Comprehensive Tests for a Form Component **1hr 15min**

## Module 15: Module 15: Integrating with TypeScript

**5 Lessons**

1. 79. Why TypeScript for React? Introduction to Static Typing Benefits **25min**

2. 80. Setting up a TypeScript-based React Project **30min**

3. 81. Typing Props, State, and Events in React Components **40min**

4. 82. Typing Custom Hooks and Context Providers **35min**

5. 83. Recap & Practical Task: Convert a Small React Project to TypeScript **1hr**

## Module 16: Module 16: Exploring the React Ecosystem: Next.js

**5 Lessons**

1. 84. Introduction to Next.js: The React Framework for Production **30min**

2. 85. Data Fetching Strategies: SSR, SSG, ISR, Client-Side **40min**

3. 86. File-based Routing and API Routes in Next.js **35min**

4. 87. Optimizing Images, Fonts, and Scripts in Next.js **30min**

5. 88. Recap & Practical Task: Build a Simple Blog with Next.js **1hr 15min**

## Module 17: Module 17: Authentication in React Applications

5 Lessons

1. 89. Token-based Authentication (JWT) Flow in React **30min**

2. 90. Creating Protected Routes and Authorization Logic **30min**

3. 91. Integrating with Authentication Services (e.g., Firebase Auth, Auth0 - Conceptual) **35min**

4. 92. OAuth 2.0 (Conceptual Overview) **20min**

5. 93. Recap & Practical Task: Implement Basic Authentication Flow in a React App **1hr**

## Module 18: Module 18: Deployment & Career Path

5 Lessons

1. 94. Building React Apps for Production: Optimization & Minification **20min**

2. 95. Deploying to Modern Platforms (Vercel, Netlify, Render - Overview) **30min**

3. 96. Environment Variables for Production **20min**

4. 97. Course Summary & Next Steps in Your React Career **20min**

5. 98. Recap & Practical Task: Deploy Your React Portfolio **45min**

## Module 19: Module 19: Capstone Project: Full-Stack Application

5 Lessons

1. 99. Project Planning, API Design (for existing Backend), and Frontend Architecture **45min**

2. 100. Building the Frontend (React) and Connecting to an API (Node.js/Express or similar) **3hr**

3. 101. Implementing Advanced Features (Authentication, State Management, Routing) **1hr 30min**

4. 102. Final Testing, Performance Checks, and Deployment Preparation **1hr**

5. 103. Recap & Final Project Submission **1hr 30min**

---

*This curriculum is subject to minor adjustments to ensure the most up-to-date and effective learning experience.*