

Learn how to use React, Redux, React Router, and modern JavaScript to build an App with React. This course will teach you how to architect and build components in React that work together to present a consistent view across you web app.

28 hours of Instructor Led Training

Hands on work and assignments along with Mentor

Master Components, JSX, State Management, Routing, Hooks and much more.

Develop rich internet applications using React and Redux

Day 1

1. JavaScript Features (ES6 and beyond)

React utilizes JavaScript syntax, most often referred to JavaScript ES6 and beyond (ES7, ES8, ES9).

- Classes
- Object Functions
- Array methods
- Arrow Functions
- Block Scope
- Default Values and rest/spread operators
- Destructuring
- Template Strings

2. Starting with React JS

- Introduction to React JS
- Smart DOM Updates
- The React Element Tree
- Tree Reconciliation
- JSX
- The React UI Workflow
- Understanding Components
- Create-react-app
- React Developer Tools
- Structuring the Application
- React's Entry Point
- The Top Level Component

3. React Component Approaches

- Four ways to create React Components
- createClass Component
- Class Component
- Function Component

- Arrow Function Component
 - Functional Component Benefits
 - When to use Class vs. Functional Components
 - Container vs. Presentational Components
4. Templating Using JSX
 - Expressions
 - Using Logical Operators
 - Specifying attributes
 - Specifying Children
 - Fragments
 - Applying CSS via className and styles.
 5. Working with state and props
 - What is state and its significance
 - Read state and set state
 - Passing data to component using props
 - Validating props using propTypes
 - Supplying default values to props using default props
 - Stateful and stateless components
 - Implementing child to parent communication using callbacks.
 6. Rendering Lists
 - Using react key prop
 - Using map function to iterate on arrays to generate elements
 7. Event Handling in React
 - Understanding React Event system
 - Understanding React Synthetic Event
 - Passing arguments to event handlers
 8. Component Lifecycle and handling errors
 - Understanding the lifecycle methods
 - Handle errors using error boundaries
 - Stateful components with lifecycle methods
 - Implementing try catch mechanism using error boundaries
 9. Working with forms
 - Controlled Components
 - Uncontrolled Components
 - Significance of default value prop
 - Using React ref property to get access to DOM element
 - Using different form controls
 10. Context
 - What and when to use Context
 - Creating context
 - Context.Provider
 - Context.Consumer

11. Code Splitting

- What is code splitting
- Why do we need code splitting
- React.lazy and Suspense
- Route based code splitting

12. Hooks

- What are hooks
- Different types of hooks
- Using State and effect hooks
- Rules of hooks
- User defined hooks

13. React Router

- Significance of Routing
- Understanding routing and SPA
- Working with BrowserRouter and HashRouter Components
- Configuring routes
- Using switch component to define routing rules
- Dynamic routes using route params
- Nested routes
- Navigating to pages using Link and NavLink Component
- Redirecting routes

14. Redux

- Introduction to Redux
- Three core Redux principles
- Redux Flow Overview
- Actions, Stores and Reducers
- Immutability and Handling Immutability
- Connecting React to Redux
- mapStateToProps
- mapDispatchToProps
- Redux Dev Tools
- Async in Redux
- Mock API
- Middleware and Async Library Options
- Thunk introduction
- Async status and Error Handling

15. Testing React

- Testing Frameworks and Libraries
- Jest
- React Testing Library

16. Production Builds

PREREQUISITES

- Basic HTML and CSS skills
- Intermediate JavaScript skills

Installation Required

- Nodejs & npm – npm will be automatically installed on node JS installation
- Visual Studio Code IDE
- Latest Chrome browser