React Essentials

Duration

3 Days

Description

Discover the world of React in our React Essentials course where you'll not only learn what React is and why it's indispensable in modern web development but also explore the foundational architecture of React applications. Dive deep into JSX and the Virtual DOM to optimize your UI development, and become proficient in creating interactive interfaces using React components. You'll not only learn React best practices, including the power of Hooks, but also master the art of creating and validating forms using controlled components. Expand your toolkit by making AJAX calls to REST APIs and configuring routing for various navigation scenarios. Finally, ensure the reliability of your code through comprehensive testing with Jest and Testing Library, making you a skilled React developer ready to tackle real-world projects.

Objectives

- Know what React is and why it is useful
- Explore the basic architecture of a React application
- Gain a deep understanding of JSX and the Virtual DOM
- Use React components to build interactive interfaces
- Learn React best practices, including Hooks
- Create and validate forms using controlled components
- Make AJAX calls to read or change via a REST API
- Configure simple and complex routing
- Test React components with Jest and Testing Library

Prerequisites

All attendees must have experience with modern JavaScript or TypeScript, including the new language features like classes, modules, arrow functions, and destructuring.

Training Materials

All students receive comprehensive courseware covering all topics in the course. Courseware is distributed via GitHub in the form of documentation and extensive code samples. Students practice the topics covered through challenging hands-on lab exercises.

Training 4 Programmers

Software Requirements

Students will need a free, personal GitHub account to access the courseware. Students will need permission to install Node.js and Visual Studio Code on their computers. Also, students will need permission to install NPM Packages and Visual Studio Extensions. If students are unable to configure a local environment, a cloud-based environment can be provided.

Outline

- React Project Setup
  - Create a new project
  - Folder Structure
  - Browser Support
  - Styles and Assets
  - Dependencies
- React Overview
  - React Overview
  - Why React?
  - Single-page web application architecture
- Component Rendering and JSX
  - What problem does JSX solve?
  - Embedding Expressions
  - Specifying Attributes
  - Using Fragments
  - Virtual DOM and Fiber Nodes
  - Ternary Operator (?)
  - Logical (&& and ||) Operators
  - Rendering a list of data
  - Optimizing rendering with keys
- Components
  - Creating an Element
  - Create a Function Component
  - Rendering a Component
  - Composing & Reuse
- Component Props
  - Immutable Props
  - String Literals vs. Expressions
  - Prop Types

- – Default Prop Values
- – Naming Patterns for Props
- Component Events
  - – What are Events?
  - – Common Events in React: Click and Change
  - – Event Handlers and Functional Component
  - – Passing Event Handlers via Props
- Component Hooks
  - – What is Component State?
  - – State Hook
  - – Effect Hook
  - – Callback Hook
  - – Custom Hooks
- Component Architecture
  - – Reusable Components
  - – Component Communication
  - – Design Patterns
  - – Container and Presentation Components
  - – Defining Prop Drilling
- Capture Data with Forms
  - – Controlled and Uncontrolled Components
  - – Enable Change Logic across Multiple Form Controls
  - – Wiring up Input, Textarea, and Select
  - – Handling different types of Input
  - – Form Validation and Submission
- AJAX and RESTful APIs
  - – Fetch API
  - – Get Data from a REST API for use with a React
  - – Loading Data on the Initial Render with the Effect Hook
  - – Add, Replace, and Remove RESTful API Data Calls
- Routing with React Router
  - – What problem does React Router solve?
  - – Installation
  - – Basics
  - – Handling Not Found (404)
  - – Parameters (Url & Query)

- Nesting
- Unit Testing Overview
  - Jest and Testing Library
  - What are React components tested for?
  - Tests, Test Suites, Assertions, and Mocking
  - Test DOM rendering
  - Test Event Handlers with Spies
  - Test Custom Hooks
  - Mocking Components
  - Mocking Hooks