Mastering React | React Foundation

COURSE OVERVIEW:

Mastering React is a comprehensive hands-on course that aims to be the single most useful resource on getting up to speed quickly with React. Geared for more experienced web developers new to React, this course provides students with the core knowledge and hands-on skills they require to build reliable, powerful React apps. After the first few modules, you'll have a solid understanding of React's fundamentals and will be able to build a wide array of rich, interactive web apps with the framework. The first module is an introduction to the new functionality in ECMAScript 6 (JavaScript). Client-side routing between pages, managing complex state and heavy API interaction at scale are also covered.

This course consists of two parts. In the first part of the course, students will explore all the fundamentals with a progressive, example-driven approach. You'll create your first apps, learn how to write components, start handling user interaction, and manage rich forms. We end the first part by exploring the inner workings of Create React App (Facebook's tool for running React apps), writing automated unit tests, and building a multi-page app that uses client-side routing.

The latter part of the course moves into more advanced concepts that you'll see used in large, production applications. These concepts explore strategies for data architecture, transport, and management:

- Redux is a state management paradigm based on Facecourse's Flux architecture. Redux provides a structure for large state trees and allows you to decouple user interaction in your app from state changes.
- GraphQL is a powerful, typed, REST API alternative where the client describes the data it needs.
- Hooks is a powerful, new way to maintain state and properties with functional components and the future of React according to Facebook.

Hands-on Learning: Every project in this course was built using Create React App. Create React App is based on Webpack, a tool that helps process and bundle our various JavaScript, CSS, HTML, and image files. We explore Create React App in-depth in the module "Using Webpack with Create React App."

- Students will build Single Page Applications (SPA), create robust routing with error handling, and both class and functional reusable components.
- The lab project will also include the use of form validation.
- The Flux and Redux parts of the lab project will assist the student in understanding the use of advanced state in React.
- The lab project will importantly unit test the application using both Jest and Enzyme.

WHO WILL BENEFIT FROM THIS COURSE?

This is an introductory level React development course but an intermediate level web development class, designed for experienced web developers that need to further extend their skills in web development.

PREREQUISITES:

To be successful in this class, incoming attendees are required to have current, hands-on experience in developing basic web applications, and be versed in HTML5, CSS3, and JavaScript.

Attendees should have experience with the topics in the following courses, or should have recently attended these as a pre-requisite:

- TT4650 Web Essentials: HTML5 / CSS3 & JavaScript

COURSE OBJECTIVES:

Working within an engaging, hands-on learning environment, attendees will learn to:

- A basic and advanced understanding of React components
- An advanced, in-depth knowledge of how React works
- A complete understanding of using Redux
- How to build, validate, and populate interactive forms
- How to use inline styles for perfect looking components
- How to test React components
- How to build and use components
- How to get control of your build process
- A deep understanding of data-driven modeling with props and state
- How to use client-side routing for pages in your apps
- How to debug a React application

COURSE OUTLINE:

ES6 Primer (Optional)

- Prefer const and let over var
- Arrow functions
- Modules
- Object.assign()
- Template literals
- The spread operator and Rest parameters
- Enhanced object literals
- Default arguments
- Destructuring assignments

Your first React Web Application

- Setting up your development environment
- JavaScript ES6 /ES7
- Getting started
- What's a component?
- Our first component
- Building the App
- Making the App data-driven
- Your app's first interaction
- Updating state and immutability
- Refactoring with the Babel plugin transform-class-properties

JSX and the Virtual DOM

- React Uses a Virtual DOM
- Why Not Modify the Actual DOM?
- What is a Virtual DOM?
- Virtual DOM Pieces
- ReactElement
- JSX
    - JSX Creates Elements
    - JSX Attribute Expressions
    - JSX Conditional Child Expressions
    - JSX Boolean Attributes
    - JSX Comments
    - JSX Spread Syntax
    - JSX Gotchas
- JSX Summary

Components

- A time-logging app
- Getting started
- Breaking the app into components
- The steps for building React apps from scratch
- Step 2: Build a static version of the app
- Step 3: Determine what should be stateful
- Step 4: Determine in which component each piece of state should live
- Step 5: Hard-code initial states
- Step 6: Add inverse data flow
- Updating timers
- Deleting timers
- Adding timing functionality
- Add start and stop functionality
- Methodology review

Advanced Component Configuration with props, state, and children

- ReactComponent
- Props are the parameters
- PropTypes
- Default props with getDefaultProps()
- context
- state
- Stateless Components
- Talking to Children Components with props.children

Forms

- Forms 101
- Text Input
- Remote Data
- Async Persistence
- Redux
- Form Modules

Unit Testing & Jest

- Writing tests without a framework
- What is Jest?
- Using Jest
- Testing strategies for React applications
- Testing a basic React component with Enzyme
- Writing tests for the food lookup app
- Writing FoodSearch.test.js

Routing

- What's in a URL?
- React Router's core components
- Building the components of react-router
- Dynamic routing with React Router
- Supporting authenticated routes

Intro to Flux and Redux

- Why Flux?
- Flux is a Design Pattern
- Flux implementations
- Redux & Redux's key ideas
- Building a counter
- The core of Redux
- The beginnings of a chat app
- Building the reducer()
- Subscribing to the store
- Connecting Redux to React

React Hooks

- Motivation behind Hooks
- How Hooks Map to Component Classes
- Using Hooks Requires react "next"
- useState() Hook Example
- useEffect() Hook Example
- useContext() Hook Example
- Using Custom Hooks

Intermediate Redux

- Using createStore() from the redux library
- Representing messages as objects in state
- Introducing threads
- Adding the ThreadTabs component
- Supporting threads in the reducer
- Adding the action OPEN_THREAD
- Breaking up the reducer function
- Adding messagesReducer()
- Defining the initial state in the reducers
- Using combineReducers() from redux

Using Webpack with Create React App

- JavaScript modules
- Create React App
- Exploring Create React App
- Webpack basics
- Making modifications
- Hot reloading; Auto-reloading
- Creating a production build
- Ejecting
- Using Create React App with an API server
- When to use Webpack/Create React App

Using GraphQL

- Your First GraphQL Query
- GraphQL Benefits
- GraphQL vs. REST
- GraphQL vs. SQL
- Relay and GraphQL Frameworks
- Chapter Preview
- Consuming GraphQL
- Exploring with GraphiQL
- GraphQL Syntax 101.
- Complex Types
- Exploring a Graph
- Graph Nodes; Viewer
- Graph Connections and Edges
- Mutations
- Subscriptions
- GraphQL With JavaScript
- GraphQL With React

SUNSET LEARNING INSTITUTE (SLI) DIFFERENTIATORS:

Sunset Learning Institute (SLI) has been an innovative leader in developing and delivering authorized technical training since 1996. Our goal is to help our customers optimize their cloud technology investments by providing convenient, high quality technical training that our customers can rely on. We empower students to master their desired technologies for their unique environments.

What sets SLI apart is not only our immense selection of trainings options, but our convenient and consistent delivery system. No matter how complex your environment is or where you are located, SLI is sure to have a training solution that you can count on!

Premiere World Class Instruction Team

- All SLI instructors have a four-year technical degree, instructor level certifications and field consulting work experience.
- Sunset Learning has won numerous Instructor Excellence and Instructor Quality Distinction awards since 2012

Enhanced Learning Experience

- The goal of our instructors during class is ensure students understand the material, guide them through our labs and encourage questions and interactive discussions.

Convenient and Reliable Training Experience

- You have the option to attend classes at any of our established training facilities or from the convenience of your home or office with the use of our HD-ILT network (High Definition Instructor Led Training)
- All Sunset Learning Institute classes are guaranteed to run – you can count on us to deliver the training you need when you need it!

Outstanding Customer Service

- Dedicated account manager to suggest the optimal learning path for you and your team
- Enthusiastic Student Services team available to answer any questions and ensure a quality training experience