React ver 16 With Redux

Outline

- Introduction
- Package Manager (npm/Yarn)
    - npm
        - What is Node.js?
        - What is npm?
        - Using npm
        - npm Scripts
        - Working with HapiJS
- Compiler Setup
    - Setup (Babel)
- ES6/ES2015
    - Classes
    - Scope (var, let, const)
    - Arrow Functions
    - Modules
    - Template Literals
    - Default, Rest, Spread
    - Default
    - Rest
    - Spread
    - Destructuring
    - Optional Parameters
    - Object.assign()
    - Object Initializer
- Project Setup (Create React App)
    - Create new Project
    - Folder Structure
    - Browser Support
    - Styles and Assets
    - Dependencies
- Best Practices (Code Organization & Conventions)
- React Overview
    - Why React?
    - What it is?
    - Why it is useful?
    - Angular, React Compared
    - Web application architectures
        - Server-side web application architecture
        - Single-page web application architecture
    - React Architecture
- Elements
    - Hello World in JavaScript
    - Hello World in React

- JSX
    - Replacing createElement
    - Embedding Expressions
    - Specifying Attributes
- Virtual DOM
- Components
    - Creating an Element
    - Create a Function Component
    - Rendering a Component
    - Creating a Class Component
    - Composing & Reuse
- Props
    - Read-only
    - String Literals vs. Expressions
    - Function vs. Class Components
- Events
    - Listening/Subscribing/Wiring to an Event
    - In Vanilla JavaScript
    - In React: Function Component
    - In React: Class Component
    - Binding
        - Why Binding is Necessary?
        - Class Method
        - Arrow Function
    - Passing Parameters
        - Using Arrow Functions
        - Using Bind
    - Handling Events
        - Using Arrow Functions
        - Using Bind
        - Synthetic Events
- State
    - Defining
    - Using State Correctly
    - Data Flows Down
    - Converting a Function Component to a Class Component
- Lifecycle
    - What are Lifecycle Methods
    - Understanding Mounting
    - Common vs. Less Common Methods
    - Using Lifecycle Methods
- Conditional Rendering
    - If, else
    - Conditional Operator (?)
    - Logical (&&) Operator
- Lists
    - In Vanilla JavaScript: for loop, array.forEach, array.map
    - In React: using Elements, Components
    - Why Keys are Needed
- Component Architecture

    - Reuse
    - Component Communication
    - Design Patterns
        - Container and Presentation Components
        - Composition vs. Inheritance
- Forms
    - Controlled Components
    - Reuse of Change Logic across Multiple Inputs
    - Handling Form Submission
    - Controlling Other Form Elements: select, textarea, number
    - Validation
    - Uncontrolled Components
- HTTP
    - Axios library
    - Fetch API
    - Using with React (HTTP GET)
    - Refactoring for Reuse
    - HTTP POST, PUT, DELETE
- Routing (React Router)
    - Installation
    - Basics
    - Handling Not Found (404)
    - Parameters (Url & Query)
    - Nesting
- Hooks
    - Defined
    - Why Hooks?
    - No Breaking API Changes
    - Hooks API
    - useState
    - useEffect
    - Custom Hooks
    - Rules of Hooks
- Build & Deploy
    - Building a React Application for Production
    - Deploying a React Application
    - Serving Apps with Client-Side Routing
    - Customizing Environment Variables
- Redux
    - What is Redux?
    - What is State?
    - Benefits Checklist
    - Principles of Redux
    - Core Concepts (Store, State, Reducers, Actions, Action Creators)
    - Complementary Packages
    - When do you need Redux?
    - Basic Redux Example (includes time traveling)
    - Gotchas/Tips
- Using Redux with React (React Redux Library)
    - The connect function

- Writing mapState functions
  - Writing mapDispatch Functions
  - Provider
  - Inside React Redux
  - Example
- Asynchronous Actions (Redux Thunk)
  - Overview
  - Async Actions (Thunks)
  - Installation
  - Your First Thunk
  - Full CRUD Example
- Putting It All Together (React & Redux & Thunk)
- Unit Testing
  - Tools (Jest, Enzyme,Mocha)
  - Syntax
  - Testing Vanilla JavaScript with Jest
  - Mocking
    - Mocking Modules
    - Mocking Functions