**Day 1 – Introduction to React & Components (2 hrs)**

**Topics Covered:**

* How React Works
* Setting up React (using create-react-app)
* React Components (Functional & Class)
* State in React
* Using React Dev Tools
* DOM Events & this binding

**Lab Exercises:**

1. Create a new React app using create-react-app.
2. Build a simple functional component that displays "Hello, React".
3. Convert the functional component to a class component and implement a state variable (e.g., a counter).
4. Add a button to update the counter using DOM events (onClick) and check changes in React Dev Tools.

**Day 2 – Props, Forms, Lists & Single Page Apps (2 hrs)**

**Topics Covered:**

* Intro to Forms in React
* Single Page Applications (SPA) concept
* Nesting Components
* Props in React
* Outputting Lists
* Stateless Components

**Lab Exercises:**

1. Create a nested component structure (Parent → Child → Grandchild).
2. Pass props from parent to child components.
3. Create a form to take user input (e.g., name & email) and display submitted values.
4. Render a list of items (e.g., shopping list) using map() with props.

**Day 3 – Conditional Rendering, Functions as Props & Virtual DOM (2 hrs)**

**Topics Covered:**

* Conditional Output (if-else, ternary)
* Functions as Props
* Virtual DOM concept
* Importing and using CSS files
* Lifecycle Methods in class components

**Lab Exercises:**

1. Create a component that conditionally renders a message based on state.
2. Pass a function as a prop from parent to child to update parent state.
3. Add CSS styling and apply conditional styles.
4. Implement componentDidMount to fetch some static data (simulate API call) and display it.

**Day 4 – Routing, Axios & Advanced Concepts (2 hrs)**

**Topics Covered:**

* Introduction to React Router
* Using <Link> & <NavLink>
* Programmatic redirects
* Higher Order Components (HOCs)
* Using Axios to fetch data
* Route Parameters & <Switch> tag

**Lab Exercises:**

1. Set up routing for three pages: Home, About, Contact.
2. Use <NavLink> for navigation with active styling.
3. Implement programmatic redirect after a form submission.
4. Create an HOC to wrap a component and display a header.
5. Use Axios to fetch data from a public API and display it on a page.
6. Use route parameters to display details of a selected item.