

React Course Overview

Page 1: React Basics

1. Introduction to React:

- JavaScript library for building user interfaces.
- Component-based architecture.
- Maintained by Meta (Facebook).

2. JSX (JavaScript XML):

- Syntax extension allowing HTML-like code in JavaScript.
- Example: `<h1>Hello, React!</h1>`

3. Components:

- Functional: `function MyComponent() {}`
- Class-based: `class MyComponent extends React.Component {}`

4. Props and State:

- Props: Read-only data passed to components.
- State: Internal data maintained by the component.

5. Event Handling:

- `onClick`, `onChange` handlers.
- Example: `<button onClick={handleClick}>Click Me</button>`

6. Conditional Rendering:

React Course Overview

- Use ternary operator or && inside JSX.
- Example: `{isLoggedIn ? <Logout /> : <Login />}`

7. Lists and Keys:

- Render multiple elements using `.map()`.
- Keys help React identify elements uniquely.

React Course Overview

Page 2: React Advanced Topics

1. Hooks:

- useState: Manage local state.
- useEffect: Side effects like API calls.
- useContext: Access context globally.

2. React Router:

- Navigate between pages/components.
- Example: `<Route path="/home" element={<Home />} />`

3. Lifting State Up:

- Share state between components by moving it to a common ancestor.

4. Forms Handling:

- Controlled components using useState.
- Handle input with onChange.

5. Project Structure Tips:

- Organize components, pages, assets.
- Example folders: /components, /pages, /assets

6. API Calls with Axios:

- Make HTTP requests.

React Course Overview

- Example: `axios.get('/api/data')`

7. Deployment Tips:

- Use Netlify, Vercel, or GitHub Pages.
- Build with ``npm run build``