Conditional Rendering in React

✉️digvijayaditya8@gmail.com

Understanding Conditional Rendering

- In React, conditional rendering allows you to dynamically decide which component or element to display based on certain conditions.

- Think of it as JavaScript's **if-else** or **ternary operators** but used for rendering UI.

Why Use Conditional Rendering?

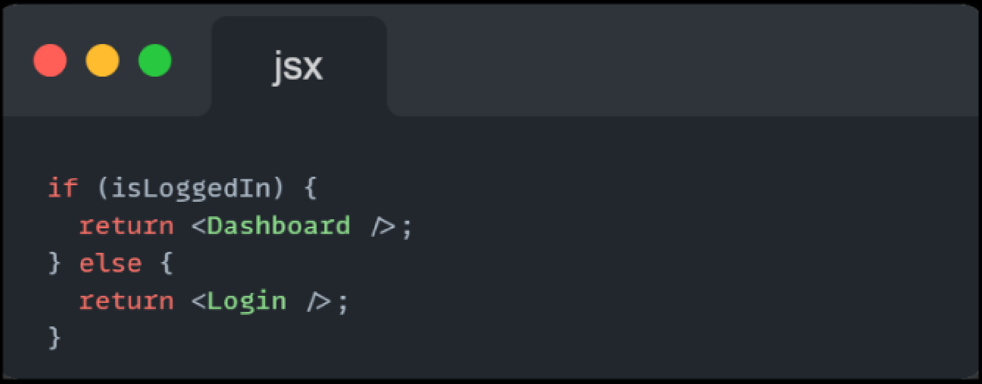
🌟 Display different UI based on user actions or app state.

🎯 Handle various scenarios like loading states, error messages, or authentication checks.

🛠️ Improve user experience by dynamically updating the interface.

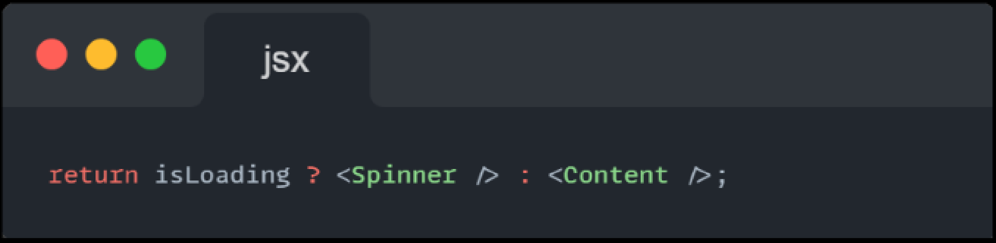
Methods for Conditional Rendering

1. **if-else Statements:**
    - Use this for complex conditions.



```jsx
if (isLoggedIn) {
  return <Dashboard />;
} else {
  return <Login />;
}
```

2. **Ternary Operators:**
    - Ideal for simple conditions.



```jsx
return isLoading ? <Spinner /> : <Content />;
```

✉️digvijayadityaa8@gmail.com

3. Logical && Operator:

- Render a component only if the condition is true.

```jsx
return showError && <ErrorMessage />;
```

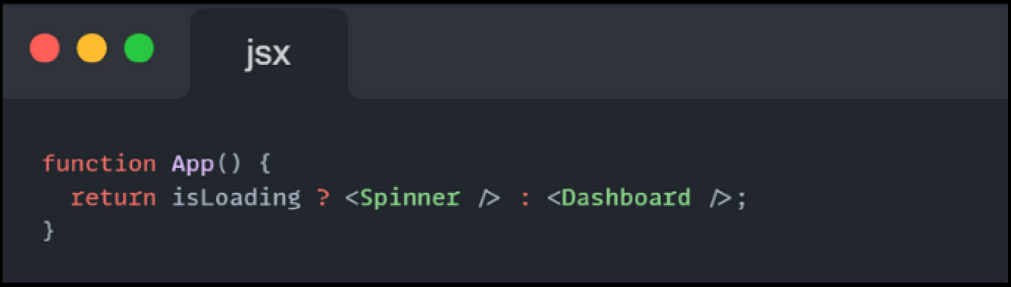
4. Switch Statements:

- Best for multiple conditions.

```jsx
switch (status) {
  case 'loading': return <Spinner />;
  case 'error': return <Error />;
  default: return <Content />;
}
```

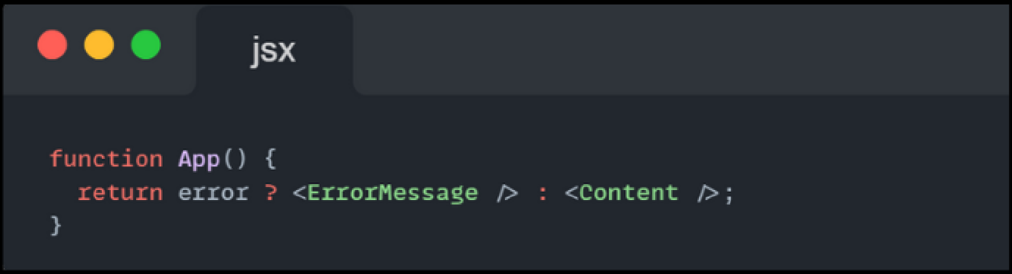
Real-World Examples

Example 1: Displaying a Loading Spinner



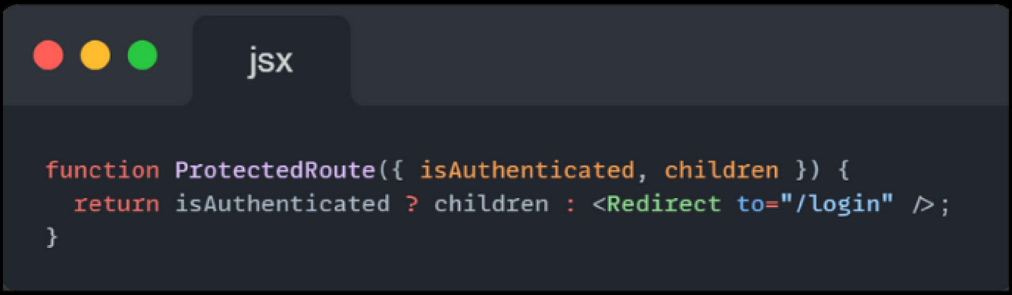
```jsx
function App() {
  return isLoading ? <Spinner /> : <Dashboard />;
}
```

Example 2: Showing an Error Message



```jsx
function App() {
  return error ? <ErrorMessage /> : <Content />;
}
```

Example 3: Protected Routes



```jsx
function ProtectedRoute({ isAuthenticated, children }) {
  return isAuthenticated ? children : <Redirect to="/login" />;
}
```

Best Practices

✅ Keep conditions simple and readable.

🧩 Use custom functions or hooks for complex logic.

🌟 Break down large components into smaller ones to manage conditional rendering effectively.

🛠️ Test thoroughly to ensure edge cases are handled.

Do you find it helpful?

Let me know down in the comments!

Digvijay Aditya

Follow For More!