

Week 2: Intermediate Concepts

Day 8: Conditional Rendering – Show/Hide Elements, If/Else in JSX

1. What is Conditional Rendering?

Conditional rendering means **showing UI based on a condition** (true/false). - Just like in plain JavaScript we use `if`, `else`, or `?:` operator. - In React, we use these conditions **inside JSX** to control what gets displayed.

2. Why do we use it?


- To show different UI based on state or props.
- Examples:
 - Show a **login button** if the user is not logged in, otherwise show **logout button**.
 - Show an **error message** only if something went wrong.
 - Hide elements when they are not needed.

3. Different Ways of Conditional Rendering

a) Using `if/else` outside JSX

```
function WelcomeMessage({ isLoggedIn }) {  
  if (isLoggedIn) {  
    return <h2>Welcome back!</h2>;  
  } else {  
    return <h2>Please log in</h2>;  
  }  
}
```

b) Using Ternary Operator inside JSX


```
function LoginStatus({ isLoggedIn }) {  
  return (  
    <div>  
      {isLoggedIn ? <p>You are logged in     </div>  
  );  
}
```

c) Using Logical AND (`&&`) for Short Conditions


- If condition is true → render element
- If false → render nothing

```
function Cart({ items }) {  
  return (  
    <div>  
      <h2>Your Cart</h2>  
      {items.length > 0 && <p>You have {items.length} items in your cart.</p>}  
    </div>  
  );  
}
```


d) Conditional CSS/Hide Elements

```
function Notification({ show }) {  
  return (  
    <div style={{ display: show ? "block" : "none" }}>  
       You have a new notification!  
    </div>  
  );  
}
```

4. Example: Toggle Show/Hide Text

```
import React, { useState } from 'react';  
  
function ToggleText() {  
  const [show, setShow] = useState(false);  
  
  return (  
    <div>  
      <button onClick={() => setShow(!show)}>  
        {show ? "Hide" : "Show"} Message  
      </button>  
      {show && <p>This is a secret message     </div>  
  );  
}  
  
export default ToggleText;
```

5. Exercise (15–20 minutes)

 Build a component called **AuthPanel** that: 1. Has a state `isLoggedIn` (default: false). 2. If `isLoggedIn` is `false` → show a button: **Login**. 3. If `isLoggedIn` is `true` → show: - A message: *"Welcome, User!"* - A button: **Logout**. 4. Clicking **Login** should set `isLoggedIn = true`. 5. Clicking **Logout** should set `isLoggedIn = false`.

Bonus Challenge: Add a small greeting like: *"Good Morning"* if current hour < 12, otherwise *"Good Evening"*.

Recap

- Use **if/else** for bigger blocks.
- Use **ternary operator** for inline small conditions.
- Use **&&** for quick conditional rendering.
- You can also hide elements with CSS (`display: none`).