

Task

Login Status Display with Short Circuiting

Create a React component that displays the user's login status using short circuiting.

Requirements:

Use short-circuiting to conditionally render the login status.

Create a boolean state variable isLoggedIn.

Instructions:

Create a new component called LoginStatus.

Use a state variable isLoggedIn to manage the login status.

Use short-circuiting to display the message "You are logged in" only if isLoggedIn is true.

Solution –

App.js

```
import React from "react";
```

```
import LoginStatus from "../LoginStatus";
```

```
function App() {  
  return (  
    <div className="App">  
      <LoginStatus />  
    </div>  
  );  
}
```

```
export default App;
```

```
JS App.js  X  # index.css  JS LoginStatus.js
login-status-app > src > JS App.js > App
1  import React from "react";
2  import LoginStatus from "../LoginStatus";
3
4  function App() {
5    return (
6      <div className="App">
7        <LoginStatus />
8      </div>
9    );
10 }
11
12 export default App;
13
```

Index.css

```
body {
  margin: 0;
  font-family: 'Arial', sans-serif;
  background-color: #f4f4f4;
  display: flex;
  justify-content: center;
  align-items: center;
  height: 100vh;
}

.login-container {
  text-align: center;
  background: white;
  padding: 2rem;
  border-radius: 12px;
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
}

h1 {
  font-size: 2rem;
```

```
color: #333;
margin-bottom: 1rem;
}
.status-message {
font-size: 1.2rem;
margin-bottom: 1.5rem;
color: #555;
}
.login-button {
padding: 0.8rem 1.5rem;
font-size: 1rem;
background-color: #4CAF50;
color: white;
border: none;
border-radius: 8px;
cursor: pointer;
transition: background-color 0.3s ease;
}
.login-button:hover {
background-color: #45a049;
}
```

```
JS App.js # index.css X JS LoginStatus.js
login-status-app > src > # index.css > h1
1  body {
2    margin: 0;
3    font-family: 'Arial', sans-serif;
4    background-color: #f4f4f4;
5    display: flex;
6    justify-content: center;
7    align-items: center;
8    height: 100vh;
9  }
10
11  .login-container {
12    text-align: center;
13    background: white;
14    padding: 2rem;
15    border-radius: 12px;
16    box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
17  }
18
19  h1 {
20    font-size: 2rem;
21    color: #333;
22    margin-bottom: 1rem;
23  }
24
25  .status-message {
26    font-size: 1.2rem;
27    margin-bottom: 1.5rem;
28    color: #555;
```

```
JS App.js # index.css X JS LoginStatus.js
login-status-app > src > # index.css > h1
23  }
24
25  .status-message {
26    font-size: 1.2rem;
27    margin-bottom: 1.5rem;
28    color: #555;
29  }
30
31  .login-button {
32    padding: 0.8rem 1.5rem;
33    font-size: 1rem;
34    background-color: #4CAF50;
35    color: white;
36    border: none;
37    border-radius: 8px;
38    cursor: pointer;
39    transition: background-color 0.3s ease;
40  }
41
42  .login-button:hover {
43    background-color: #45a049;
44  }
45
```

LoginStatus.js

```
import React, { useState } from "react";

const LoginStatus = () => {
  // State variable to track login status
  const [isLoggedIn, setIsLoggedIn] = useState(false);

  return (
    <div className="login-container">
      <h1>Login Status</h1>
      { /* Short-circuit rendering */ }
      {isLoggedIn && <p className="status-message">You are logged in!</p>}
      {!isLoggedIn && <p className="status-message">You are not logged
in.</p>}}

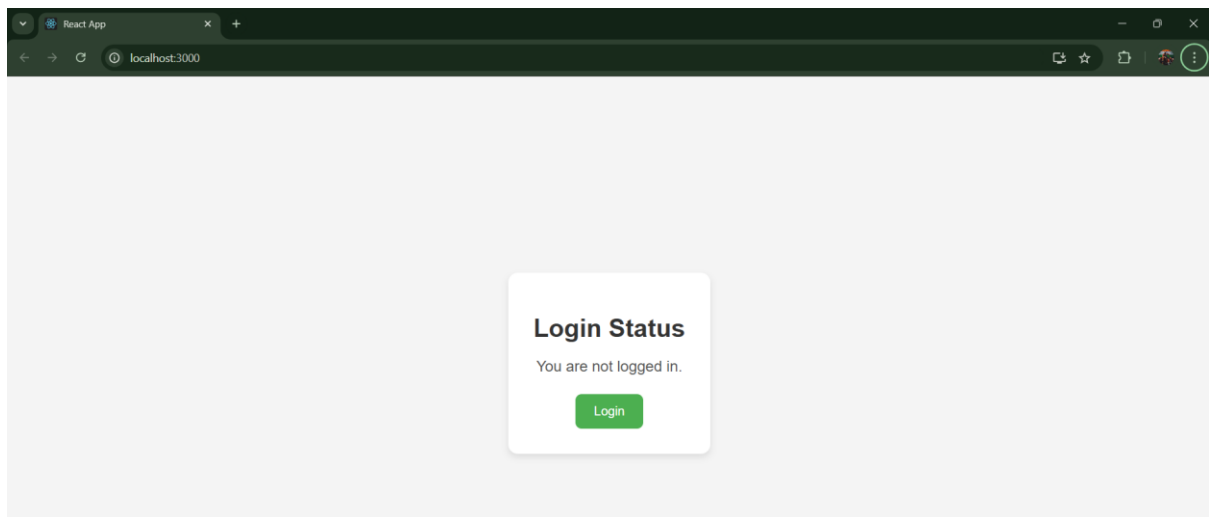
      { /* Button to toggle login status */ }
      <button
        className="login-button"
        onClick={() => setIsLoggedIn(!isLoggedIn)}
      >
        {isLoggedIn ? "Logout" : "Login"}
      </button>
    </div>
  );
};

export default LoginStatus;
```

```
JS App.js # index.css JS LoginStatus.js X
login-status-app > src > JS LoginStatus.js > [e] LoginStatus
1 import React, { useState } from "react";
2
3 const LoginStatus = () => {
4   // State variable to track login status
5   const [isLoggedIn, setIsLoggedIn] = useState(false);
6
7   return (
8     <div className="login-container">
9       <h1>Login Status</h1>
10      /* Short-circuit rendering */
11      {isLoggedIn && <p className="status-message">You are logged in!</p>}
12      {!isLoggedIn && <p className="status-message">You are not logged in.</p>}
13
14      /* Button to toggle login status */
15      <button
16        className="login-button"
17        onClick={() => setIsLoggedIn(!isLoggedIn)}
18      >
19        {isLoggedIn ? "Logout" : "Login"}
20      </button>
21    </div>
22  );
23 };
24
25 export default LoginStatus;
26
```

OUTPUT –

Before Clicking the Login Button



After Clicking Login Button

