

## 소프의 처음 만나 리액트



## 제 9장 조건부 렌더링





## Q & A

- 언제라도 질문하세요  
모르면 외우면 되지만 코딩 안되면 꼭 질문
  1. 대면수업에서 질문
  2. 한성e-class 질의응답게시판
- 강사의 1번 선생님은 여러분의 질문





# 목차

1. 조건부 렌더링이란
2. 엘리먼트 변수
3. 인라인 조건
4. 컴포넌트 렌더링 막기
5. 실습: 로그인 여부를 나타내는 툴바 만들기
6. 마치며



## 교재와 react 사이트 내용비교

3장  
4장  
5장  
6장  
8장  
9장  
10장  
11장  
12장  
13장

### MAIN CONCEPTS ^

1. Hello World
2. Introducing JSX
3. Rendering Elements
4. Components and Props
5. State and Lifecycle
6. Handling Events
7. Conditional Rendering
8. Lists and Keys
9. Forms
10. Lifting State Up
11. Composition vs Inheritance
12. Thinking In React

- <https://17.reactjs.org/docs/getting-started.html>
- 교재는 react사이트 내용을 충실히 다루고 있음
- 차이점은 각 chapter의 실습, 15장 스타일링, 16장 미니블로그
- 7장 hook는 별도
- 14장 컨텍스트는 별도

학생> 코드는 v18인데 doc은 v17 ?

교수> 심교수도 소플님의 교재내용이 react를 체계적으로 이해하기 좋음

### API REFERENCE v

HOOKS v

TESTING v

### ADVANCED GUIDES ^

Accessibility

Code-Splitting

Context

Error Boundaries



## 9.1 조건부 렌더링 (274.jsx)

- 조건부 렌더링은 어떠한 조건에 따라서 렌더링이 달라지는 것
- Codepen 사이트에서 실행 안되서 **index.js**로 변경하여 수행  
isLoggedIn 값을 **false**에서 **true**로 변화시켜봄

학생> 274.jsx를 왜 index.js로 변경하나요?

교수> index.js는 npm start에서 시작파일인데

274.jsx는 21번줄에 root.render가 있어서

기존 index.js의 내용 변경하여 실행함

```
1  import ReactDOM from 'react-dom/client';
2
3  function UserGreeting(props) {
4    |   return <h1>Welcome back!</h1>;
5  }
6
7  function GuestGreeting(props) {
8    |   return <h1>Please sign up.</h1>;
9  }
10
11 function Greeting(props) {
12   |   const isLoggedIn = props.isLoggedIn;
13   |   if (isLoggedIn) {
14   |     |   return <UserGreeting />;
15   |   }
16   |   return <GuestGreeting />;
17 }
18
19 const root = ReactDOM.createRoot(document.getElementById('root'));
20 // Try changing to isLoggedIn={true}:
21 root.render(<Greeting isLoggedIn={false} />);
22
```



## 9.2 엘리먼트 변수 ( 277Class.jsx, 277Func.jsx )

- Codepen 사이트에서 실행 안되서 index.js로 변경하여 수행
- React 사이트에서 LoginControl 클래스컴포넌트이지만 교재는 함수 컴포넌트로 작성됨
- 엘리먼트를 저장하기 위해 변수를 사용할 수 있음
- 49번줄에서 button(엘리먼트변수)에 컴포넌트에서 생성된 엘리먼트 저장
- 57번줄에서 엘리먼트 반환

```
47   let button;
48   if (isLoggedIn) {
49     button = <LogoutButton onClick={handleLogoutClick} />;
50   } else {
51     button = <LoginButton onClick={handleLoginClick} />;
52   }
53
54   return (
55     <div>
56       <Greeting isLoggedIn={isLoggedIn} />
57       {button}
58     </div>
59   )
```



## 9.3 인라인조건

- 조건문을 코드안에 집어 넣는 것
- 인라인 If
- 인라인 If-Else



## 9.3 인라인 If (279.jsx)

- **&&** 연산자를 JSX코드 안에 중괄호를 사용하여 직접 넣어서 쓰는 방식

```
1   return (  
2     <div>  
3       { count && <h1>Messages: {count}</h1>}  
4     </div>  
5   );
```





## 9.3 인라인 If-Else (281.jsx)

- 삼항연산자 사용 : 조건문 ? 참일경우 : 거짓일 경우

```
1  return (  
2    <div>  
3      The user is <b>{isLoggedIn ? 'currently' : 'not'}</b> logged in.  
4    </div>  
5  );
```



## 9.4 컴포넌트 렌더링 막기 ( 283Class.jsx, 283Func.jsx )

- 컴포넌트 렌더링을 막으려면 `null` 을 리턴하면 됨
- Codepen 사이트에서 실행 안되서 `index.js`로 변경하여 수행
- React 사이트에서 클래스컴포넌트이지만 교재는 함수 컴포넌트로 작성됨
- 6번 `Null`이 `return`되면 `WarningBanner` 컴포넌트가 `rendering`안됨



Show

Warning!

Hide

```
4 function WarningBanner(props) {  
5   if (!props.warn) {  
6     return null;  
7   }  
8  
9   return (  
10    <div className="warning">  
11      Warning!  
12    </div>  
13  );  
14 }
```

```
16 function Page(props) {  
17   const [showWarning, setShowWarning] = useState(false);  
18  
19   const handleToggleClick = () => {  
20     setShowWarning(prevShowWarning => !prevShowWarning);  
21   }  
22  
23   return (  
24     <div>  
25       <WarningBanner warn={showWarning} />  
26       <button onClick={handleToggleClick}>  
27         {showWarning ? 'Hide' : 'Show'}  
28       </button>  
29     </div>  
30   )  
}
```



## 9.5(실습) 로그인 여부를 나타내는 툴바 만들기

- 코드위치 : LandingPage.jsx, Toolbar.jsx
- 실행해봄
- Toolbar.jsx 20번줄 : 인라인 If ( 논리 AND 연산자 && )
- Toolbar.jsx 22번줄 : 인라인 If-Else ( 삼항연산자 ? T : F )



# 이제 나는 할 수 있다

1. 나는 **React**사이트에서 교재내용을 볼 수 있다.
  2. 나는 조건부 렌더링을 할 수 있다
  3. 나는 변수에 엘리먼트를 저장할 수 있다
  4. 나는 인라인조건인 **&&**와 삼항연산자( **? T : F** )를 사용할 수 있다
  5. 나는 컴포넌트 렌더링을 막을 수 있다
- 유튜브 강의 동영상 : <https://www.youtube.com/c/소플TV>
  - Q&A 사이트 : <https://www.frontoverflow.com>