

소프의 처음 만나 리액트



제 3장 JSX 소개



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. JSX란
2. JSX의 역할
3. JSX의 장점
4. JSX 사용법
5. 실습 : JSX 코드 작성해보기
6. 마치며



3.1 JSX란?

- JSX : A syntax extension to JS
- JS와 다른 새로운 언어 : JS + XML /HTML 이 교재에서는 JS+HTML

학생> 교수님~ React는 JS 라이브러리인데 왜 JSX 언어로 코딩하나요?

교수> 3.3 절에 JSX의 장점이 나옵니다.

1. 코드가 간결해진다
jsx 사용
`<div>Hello, {name}</div>`
jsx 미사용
`React.createElement('div', null, 'Hello, ${name}');`
2. 가독성이 향상
코드 파악이 쉬워서 버그 쉽게 발견할 수 있다
3. Injection Attack 해킹을 방어할 수 있다

➔ 이 교재에서는 JS 대신 모두 JSX 로 코딩



3.5 JSX 코드 작성해 보기

- 과제6에서 CRA방식으로 my-app을 만들었습니다.
- 공지사항게시판 본문코드에서 chapter_03폴더를 복사하여 c:\ws_react\my-app\src\chapter_03을 만듭니다.
- index.js 파일을 편집하여 만든다
7번 라인 추가
12번 라인 App => Library 변경

```
1  import React from 'react';
2  import ReactDOM from 'react-dom/client';
3  import './index.css';
4  import App from './App';
5  import reportWebVitals from './reportWebV
6
7  const root = ReactDOM.createRoot(document
8  root.render(
9    <React.StrictMode>
10     <App />
11   </React.StrictMode>
12 );
13
14 // If you want to start measuring perform
15 // to log results (for example: reportWeb
16 // or send to an analytics endpoint. Learn
17 reportWebVitals();
18
```

```
1  import React from 'react';
2  import ReactDOM from 'react-dom/client';
3  import './index.css';
4  import App from './App';
5  import reportWebVitals from './reportWebVita
6
7  import Library from './chapter_03/Library';
8
9  const root = ReactDOM.createRoot(document.ge
10 root.render(
11   <React.StrictMode>
12     <Library />
13   </React.StrictMode>
14 );
15
16 // If you want to start measuring performanc
17 // to log results (for example: reportWebVit
18 // or send to an analytics endpoint. Learn m
19 reportWebVitals();
20
```

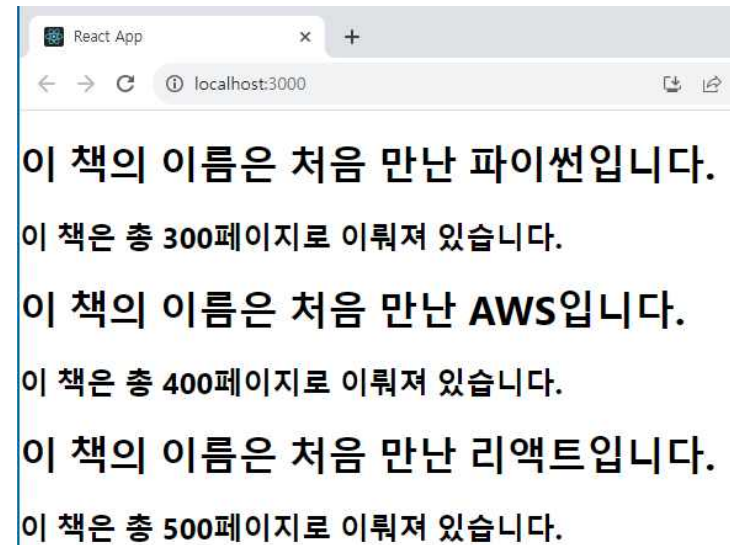


3.5 JSX 코드 작성해 보기

- VS code로 chapter_03의 Book.jsx와 Library.jsx를 봅니다
- 이 코드를 VS code terminal에서 수행해 봅니다.
C:\ws_react > cd my-app
C:\ws_react\my-app> npm start
- Index.js, Libray.jsx, Book.jsx로 만든 화면입니다

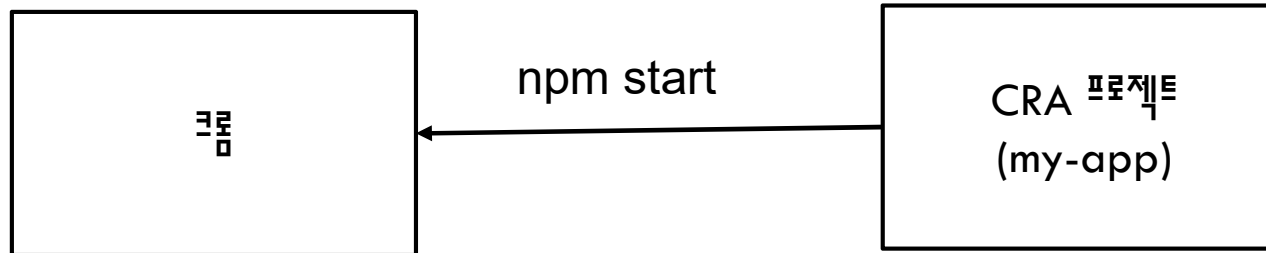
학생> 교수님~ 2장에서는 CRA방식으로 react웹을 수행했는데 3장부터 16장까지 jsx로 코딩하려니 1학년 웹기초 배운것과 전혀 연결이 안되어 코드이해하기 어렵습니다.

교수> 그래서 CDN방식으로 먼저 3장~5장 배웠고 동일한 내용을 CRA 방식으로 코딩하겠습니다. 이제부터는 JSX코딩합니다.





학생> index.html이 업로드돼요?



교수> CRA 프로젝트 코드와 npm start를 통해 만들어져서 크롬에 전달된 코드를 비교해보겠습니다.

학생> 크롬에 전달된 코드를 어떻게 보죠?

교수> 크롬> 오버튼 > 페이지소스보기

index.html과 bundle.js로 구성되어 있음

두개의 파일을 c:\ws_react\CDN\chapter_03 6주차 폴더 생성후
파일 저장합니다. (index.htm파일 만드는 방법을 시연)

학생> 3장 CRA방식인데 왜 CDN폴더에 저장하나요?

교수> index.html은 LiveServer로 실행하기 때문입니다.

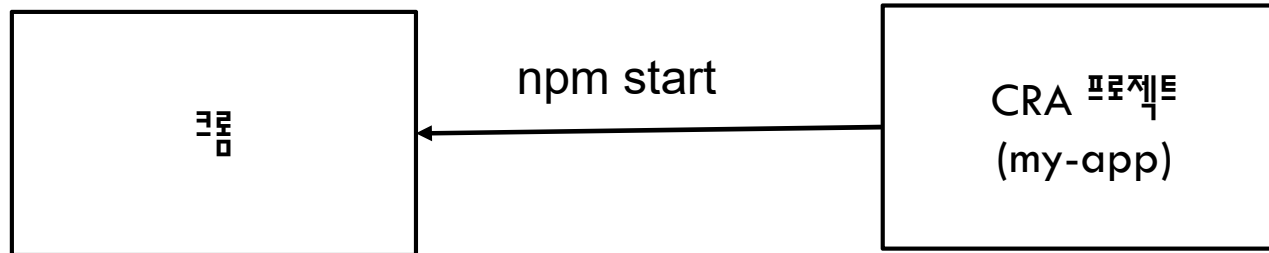
npm start는 src\index.js를 실행합니다

CDN폴더의 내용은 LiveServer로 실행하고

my-app 폴더의 내용은 npm start로 실행합니다.



학생> bundle.js가 4만줄이 넘는데요



교수> CRA 프로젝트 코드에서 npm start를 통하여 index.html과 bundle.js가 크롬에 전달됩니다.

bundle.js 파일을 분석하겠습니다.

학생> 4만줄이 넘는데 어떻게 분석하나요?

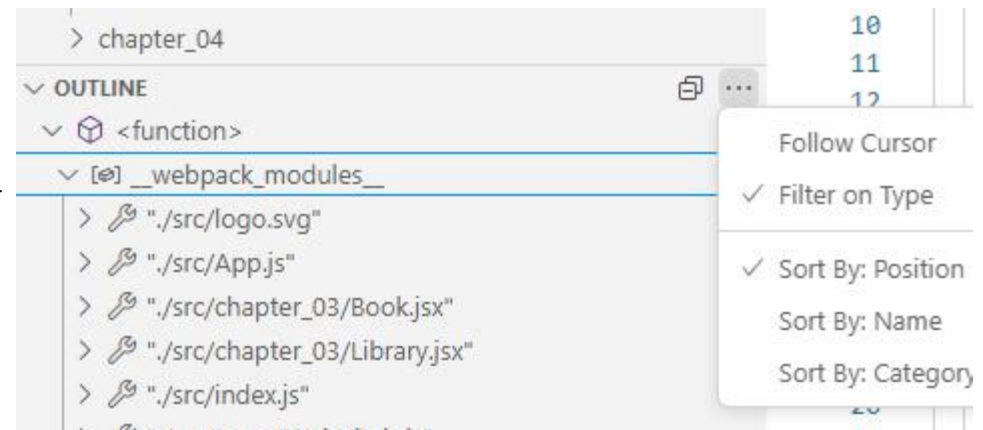
교수> vscode의 outline을 사용하겠습니다. Outline ... 옵션에서 Sort By:Position을 선택

학생> my-app에 있는

Index.js, Libray.jsx, Book.jsx 가 모두 있네요

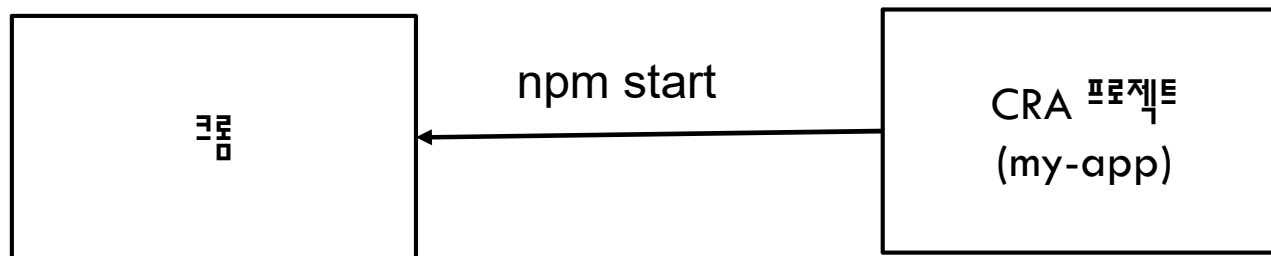
교수> index.js부터 보겠습니다

학생> my-app에 있는 index.js와 유사하네요





Babel과 Webpack



교수> Library.jsx를 보겠습니다.

학생> Library.jsx를 보니 Babel (babeljs.io)을 통해 JS로 변환되었네요.

npm start 동작에서 Babel이 수행되네요.

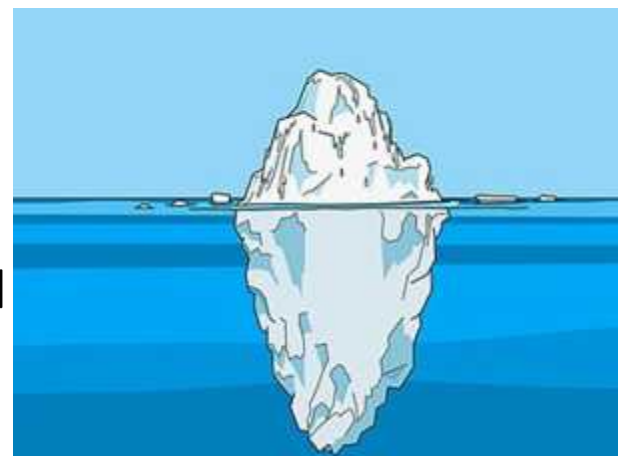
학생> 1번줄부터 해서 webpack 단어가 많이 나오네요.

교수> Index.js, Libray.jsx, Book.jsx가 bundle.js 로

합치는 역할이 webpack (webpack.js.org)이 수행

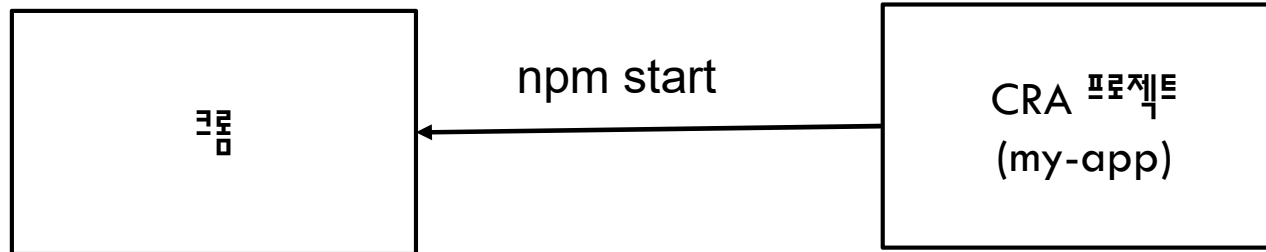
교수> npm start가 빙산의 일각처럼 Babel, webpack등을
이용하여

my-app의 JSX코드들을 index.html과 bundle.js로 변환하여
크롬으로 보내줍니다.

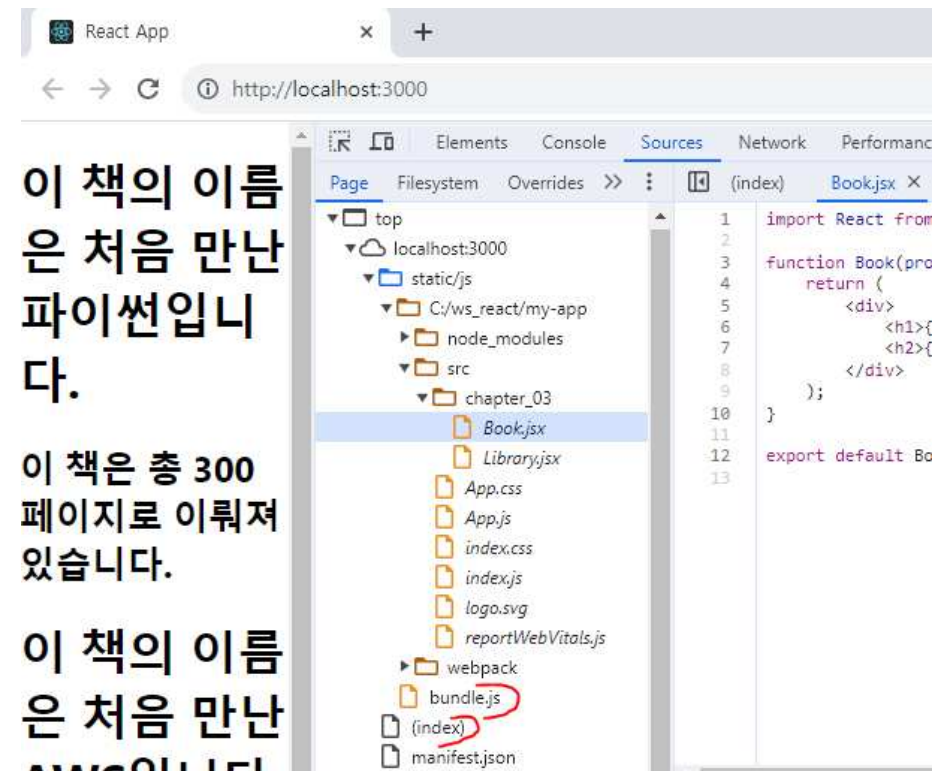




학생> npm start의 F12에서 my-app 파일이 다 보여요



교수> F12에서 my-app파일이 모두 보입니다
야호~





휴식

1. 나는 CRA방식으로 3.5 실습을 수행해보았다
2. index.js, Library.jsx, Book.jsx 한줄씩 이해 예정



학생> 교수님~ 어려워요. 코드 한줄씩 설명해주세요.

교수> (index.js), Library.jsx, Book.jsx 자세히 설명하겠습니다

학생> index.js의 11번줄 <React.StrictMode>와 12번줄 <Library />는 JS 문법이 아닌데요

교수> JSX문법입니다. 그래서 babel에서 변환해보았습니다

```
root.render(React.createElement(React.StrictMode, null, React.createElement(Library, null)));
```

React.createElement(

type, // 자료형은 HTML tag 또는 리액트 Component

[props], // 자료형은 객체(교재 36P) 또는 null 또는 blank

[...children] // blank 또는 null 또는 여러 개의 element 또는 string

)

= type이 HTML tag인 경우

- 코드위치: c:\ws_react\CDN\chapter_05_children\Child1.js

```
React.createElement( "div", {name: "소플"})
```

= type이 리액트 Component인 경우

- 코드위치: c:\ws_react\CDN\chapter_03_reactJS\index.js

```
root.render(React.createElement(Library));
```

학생> JSX이었네요. React.StrictMode는 무엇인가요?



학생> 교수님~ 어려워요. 코드 한줄씩 설명해주세요.

교수> (index.js), Library.jsx, Book.jsx 자세히 설명하겠습니다

학생> JSX이었네요. React.StrictMode는 무엇인가요?

```
root.render(React.createElement(React.StrictMode, null, React.createElement(Library, null)));
```

교수> react 사이트에서 검색하면

StrictMode is a tool for highlighting potential problems in an application.

학생> index.js에서는 Library 컴포넌트를 props는 null로 불러서 element를 만드는 동작을 하네요.

교수> 네.



학생> 교수님~ 어려워요. 코드 한줄씩 설명해주세요.

교수> index.js, (Library.jsx), Book.jsx 자세히 설명하겠습니다

학생> Libray.jsx는 아래와 같이 크롬 화면에 <div> tag와 컴포넌트인 <Book> tag를 만든다는 뜻이네요.

교수> 네. <Book> tag는 HTML tag가 아니므로 Book 컴포넌트에서 HTML tag를 만듭니다.

Book 컴포넌트 props는 name="처음 만난 파이썬" numOfPage={300} 입니다.

```
<div>
```

```
  <Book name="처음 만난 파이썬" numOfPage={300} />
```

```
  <Book name="처음 만난 AWS" numOfPage={400} />
```

```
  <Book name="처음 만난 리액트" numOfPage={500} />
```

```
</div>
```

```
React.createElement(
```

```
  type, // 자료형은 HTML tag 또는 리액트 Component
```

```
  [props], // 자료형은 객체(교재 36P) 또는 null 또는 blank
```

```
  [...children] // blank 또는 null 또는 여러 개의 element 또는 string
```

```
)
```



학생> 교수님~ 어려워요. 코드 한줄씩 설명해주세요.

교수> index.js, Library.jsx, (Book.jsx) 자세히 설명하겠습니다

학생> Book 컴포넌트에서는 아래와 같이 크롬 화면에 <div> tag와 <h1> , <h2> tag를 만든다는 뜻이네요.

교수> 네. Book.jsx에서는 HTML tag만 있고 리액트 component는 없습니다.

Book 컴포넌트 props는 name="처음 만난 파이썬" numOfPage={300} 입니다.

이었기에 크롬 화면에는

```
<div>
```

```
  <h1>
```

```
  <h2>
```

```
</div>
```

가 표시됩니다.

```
Function Book(props) {
```

```
  return (
```

```
    <div>
```

```
      <h1>{'이 책의 이름은 ${props.name}입니다.'}</h1>
```

```
      <h2>{'이 책은 총 ${props.numOfPage}페이지로 이뤄져 있습니다.'}</h2>
```

```
    </div>
```

```
  );
```

```
}
```

```
<!DOCTYPE html>
<html lang="en">
  <head>...</head>
  <body>
    <noscript>You need to enable JavaScript to run this app.</noscript>
    <div id="root">
      <div>
        <div == $0
          <h1>이 책의 이름은 처음 만난 파이썬입니다.</h1>
          <h2>이 책은 총 300페이지로 이뤄져 있습니다.</h2>
        </div>
      </div>
    </div>
  </body>
</html>
```



학생> 교수님~ 어려워요. 코드 한줄씩 설명해주세요.

교수> index.js, Library.jsx, (Book.jsx) 자세히 설명하겠습니다

학생> Book 컴포넌트를 JSX로 하니 F12 elements창과 1:1 매칭이 되어 JS보다 직관적입니다.

// Book.jsx

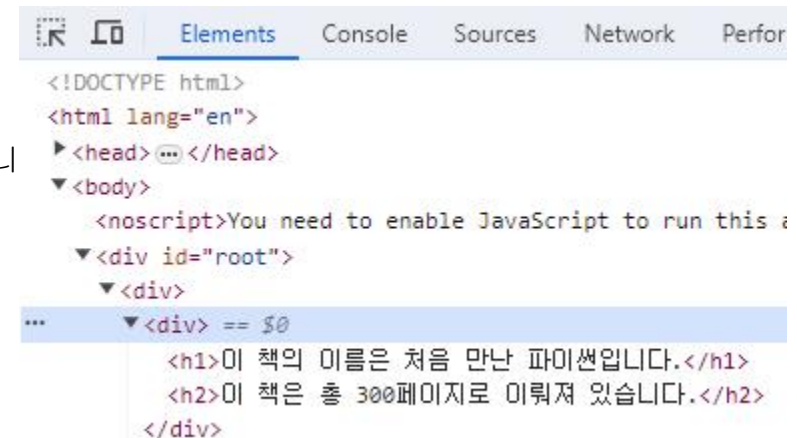
```
function Book(props) {  
  return (  
    <div>  
      <h1>{'이 책의 이름은 ${props.name}입니다.'}</h1>  
      <h2>{'이 책은 총 ${props.numOfPage}페이지로 이뤄져 있습니<br>  
</div>  
    );  
  }  
}
```

// Book.js

```
function Book(props) {  
  return React.createElement("div", null,  
    React.createElement("h1", null, `이 책의 이름은 ${props.name}입니다.`),  
    React.createElement("h2", null, `이 책은 총 ${props.numOfPage}페이지로 이뤄져 있습니다.`));  
}
```

교수> SPA 방식이기에 index.html에 root div 태그 1개만 있기에 나머지 태그는 JSX에서 만들어냅니다.

JSX코드가 F12 elements창과 1:1 매칭이 되어 코딩하기 좋습니다





학생> 교수님~ 어려워요. 코드 한줄씩 설명해주세요.

교수> index.js, Library.jsx, (Book.jsx) 자세히 설명하겠습니다

Library.jsx에서 Book 컴포넌트 호출

```
<Book name="처음 만난 파이썬" numOfPage={300} />
```

Book.jsx에서 props 전달받는 방식

```
function Book(props) {
```

```
...
```

```
<h1>{'이 책의 이름은 ${props.name}입니다.'}</h1>
```

```
<h2>{'이 책은 총 ${props.numOfPage}페이지로 이뤄져 있습니다.'}</h2>
```

```
...
```

```
}
```

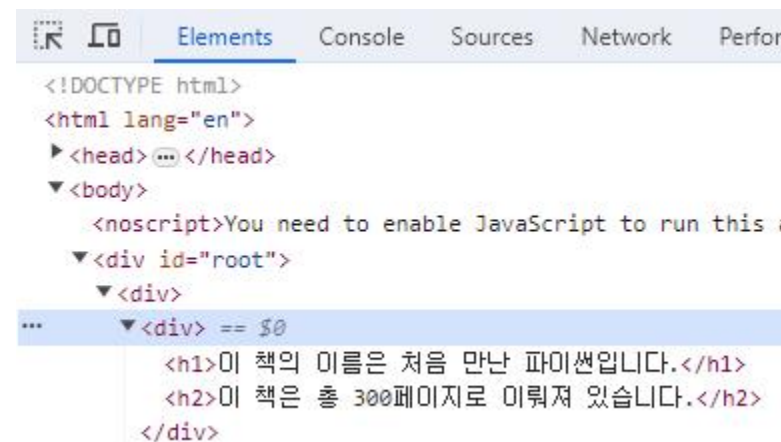
학생> Book.jsx에서 <h1>{ 는 무엇인가요?

교수> jsx에서 html 코드 사용하다가 중간에 js를 사용하고 싶으면 { 를 사용합니다.

즉 Babel은 JSX가 JS+HTML로 되어 있기에 JS모드로 분석하다가 < 태그를 만나면 HTML 모드로 분석합니다.

다시 JS모드로 분석하게 하려면 { 을 사용합니다.

```
React.createElement("h1", null, `이 책의 이름은 ${props.name}입니다.`)
```





학생> 교수님~ 어려워요. 코드 한줄씩 설명해주세요.

교수> index.js, Library.jsx, (Book.jsx) 자세히 설명하겠습니다?

Library.jsx에서 Book 컴포넌트 호출

```
<Book name="처음 만난 파이썬" numOfPage={300} />
```

Book.jsx에서 props 전달받는 방식

```
function Book(props) {
```

```
...
```

```
<h1>{'이 책의 이름은 ${props.name}입니다.'}</h1>
```

```
<h2>{'이 책은 총 ${props.numOfPage}페이지로 이뤄져 있습니다.'}</h2>
```

```
...
```

```
}
```

학생> `이 책의 이름은 \${props.name}입니다.` 이 js코드는 무슨 뜻인가요?

교수> 코드위치: c:\ws_react\CDN\chapter_03_6주차\index_Book.html

을 만들어서 js코드 이해할 예정입니다.

먼저 ES6등 JS코드 이해를 위해 JS 코드 작성해서 수행하기 위한 견본 파일 index_template.html을 만듭니다

크롬에 출력하려면 document.write()를 사용하고 F12 console창에 출력하려면 console.log()를 사용합니다.

index_Book.html 처럼 <script>에 JS코드 작성하여 수행합니다.



학생> 교수님~ 어려워요. 코드 한줄씩 설명해주세요.

교수> index.js, Library.jsx, (Book.jsx) 자세히 설명하겠습니다

Library.jsx에서 Book 컴포넌트 호출

```
<Book name="처음 만난 파이썬" numOfPage={300} />
```

Book.jsx에서 props 전달받는 방식

```
function Book(props) {  
  ...  
  <h1>{'이 책의 이름은 ${props.name}입니다.'}</h1>  
  <h2>{'이 책은 총 ${props.numOfPage}페이지로 이뤄져 있습니다.'}</h2>  
  ...  
}
```

학생> `이 책의 이름은 \${props.name}입니다.` 이 js코드는 무슨 뜻인가요?

교수> 코드위치: c:\ws_react\CDN\chapter_03_6주차\index_Book.html

- ` (Back Tic라고 말함 : 컴퓨터자판에서 Tab키 위에 있음) : Template Literals : https://www.w3schools.com/js/js_string_templates.asp
ES6에 추가됨.

- `` 은 string을 만듦

학생> string은 single 또는 double quote 을 사용하여 만드는데요

교수> ES6부터는 백틱을 사용하여 만들수도 있음 .

ES6부터 string 만드는 3가지 : ` , " , `

교수> 백틱내부에서는 \${props.name} 형태로 변수 사용가능



학생> 교수님~ 어려워요. 코드 한줄씩 설명해주세요.

교수> index.js, (Library.jsx), Book.jsx 자세히 설명하겠습니다

Library.jsx에서 Book 컴포넌트 호출

```
<Book name="처음 만난 파이썬" numOfPage={300} />
```

Book.jsx에서 props 전달받는 방식

```
function Book(props) {
```

```
...
```

```
  <h1>{'이 책의 이름은 ${props.name}입니다.'}</h1>
```

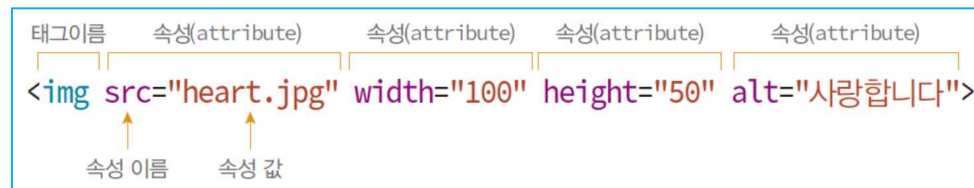
```
  <h2>{'이 책은 총 ${props.numOfPage}페이지로 이뤄져 있습니다.'}</h2>
```

```
...
```

```
}
```

학생> Library.jsx에서 <Book name="처음 만난 파이썬" numOfPage={300} /> 의 attribute가 props로
props = {name: "처음 만난 파이썬", numOfPage: 300} 로 변환되어 Book(props)로 전달되는 것은 알겠는데

{300} 괄호는 왜 있나요?





학생> 교수님~ 어려워요. 코드 한줄씩 설명해주세요.

교수> index.js, (Library.jsx), Book.jsx 자세히 설명하겠습니다

Library.jsx에서 Book 컴포넌트 호출

```
<Book name="처음 만난 파이썬" numOfPage={300} />
```

Book.jsx에서 props 전달받는 방식

```
function Book(props) {
```

```
  <h1>{'이 책의 이름은 ${props.name}입니다.'}</h1>
```

교수> C:\ws_react\my-app\src\chapter_03에

Library2.jsx와 Library3.jsx를 추가하였습니다

Library2.jsx를 실행하면

```
<Book name=" 처음 만난 리액트 " numOfPage= 500 }
```

syntax error 발생합니다

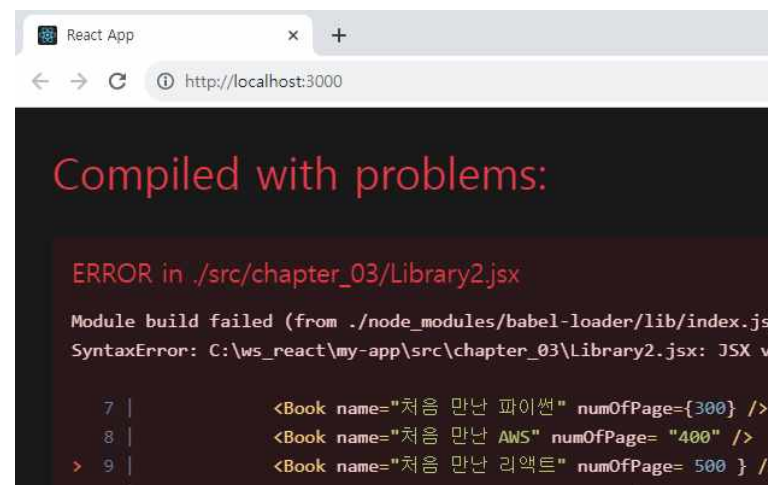
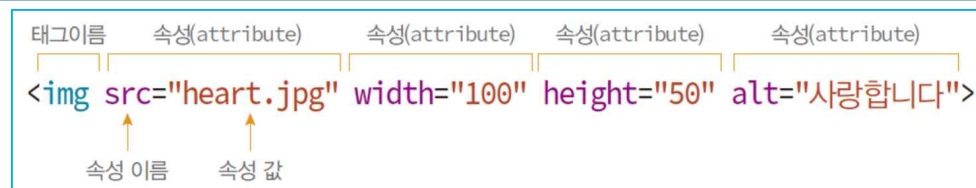
Library3.jsx를 실행하면

```
<Book name="처음 만난 파이썬" numOfPage={300} />
```

```
<Book name="처음 만난 AWS" numOfPage="400" />
```

JSX파일의 HTML에서 string이 아닌 숫자값으로 전달할때는

400 이 아니라 {400} 사용



```
▶ props: {name: '처음 만난 파이썬', numOfPage: 300}
▶ props: {name: '처음 만난 AWS', numOfPage: '400'}
```

학생> Book.jsx 4번줄 중단점에서 같은 내용이 2번 나오나요?

교수> <https://react.dev/reference/react/StrictMode#reference> 에서 StrictMode시 double rendering함



이제 나는 할 수 있다

1. 나는 3장의 JSX코드(index.js, Library.jsx, Book.jsx)를 한줄 한줄 모두 이해했다
2. 나는 JSX코드가 F12 elements창과 1:1 매칭이 되어 코딩하기 좋다
3. 나는 JSX는 JS와 HTML이 혼합되어 있기에 HTML모드에서 { }를 사용하여 JS 사용하는 방법을 알았다
즉 Book.jsx에서 JS모드에서 < 을 만나면 HTML모드시작 HTML모드에서 { 만나면 JS 모드시작 } 만나면 JS모드 종료 HTML 모드 > 만나면 HTML모드 종료 JS모드 시작
4. 나는 Library.jsx에서 HTML attribute가 리액트 컴포넌트 props로 어떻게 전달되는지 알았다.
특히 숫자값 전달방법
5. 나는 JSX코드를 F12로 디버깅하여 변수를 보는 방법을 알고 있다
6. 나는 JSX코드가 애매하면 Library2.jsx나 Library3.jsx를 만들어 돌려보고 명확한 내용을 파악할 수 있다
7. 나는 ES6등 JS학습을 위해 index_template.html파일에 index_Book.html 처럼 <script> 태그에 JS코드 작성하여 수행하고 있다.

- 유튜브 강의 동영상 : <https://www.youtube.com/c/소플TV>
- Q&A 사이트 : <https://www.frontoverflow.com>