

2. 리액트 시작하기(실습)

- 웹 사이트에 리액트 추가하기
 - create-react-app

3. JSX

3.1 JSX란?

- A syntax extension to JavaScript
- 자바스크립트의 확장 문법
 - JavaScript + XML/HTML
- 간단한 예
 - JS코드(왼쪽)에 HTML코드(오른쪽)가 결합된 모습

```
const element = <h1>Hello, world</h1>;
```

3.2 JSX의 역할

- JSX는 내부적으로 XML/HTML 코드를 JS코드로 변환하는 과정을 거침
- 우리가 JSX로 작성을 해도 최종적으로는 JS코드로 변환됨
- JSX 코드를 JS 코드로 변환시키는 함수
 - `React.createElement()`

3.2 JSX의 역할

예를 들어 다음의 JSX로 작성된 코드는

```
class Hello extends React.Component {  
  render() {  
    return <div>Hello {this.props.toWhat}</div>;  
  }  
}  
  
const root = ReactDOM.createRoot(document.getElementById('root'));  
root.render(<Hello toWhat="World" />);
```

아래처럼 JSX를 사용하지 않은 코드로 컴파일될 수 있습니다.

```
class Hello extends React.Component {  
  render() {  
    return React.createElement('div', null, `Hello ${this.props.toWhat}`);  
  }  
}  
  
const root = ReactDOM.createRoot(document.getElementById('root'));  
root.render(React.createElement(Hello, {toWhat: 'World'}, null));
```

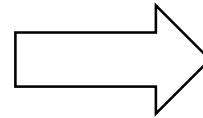
3.2 JSX의 역할

- JSX를 사용한 코드(내부적으로는 밑에 코드처럼 변환됨)

```
const element = (  
  <h1 className="greeting">  
    Hello, world!  
  </h1>  
);
```

- JSX를 사용하지 않은 코드

```
const element = React.createElement(  
  'h1', // type: HTML 태그 또는 리액트 컴포넌트  
  { className: 'greeting' }, // [props]: 속성들  
  'Hello, world!' // [...children]: 자식 엘리먼트들  
);
```



- React element(일반 객체)

```
const element = {  
  type: 'h1',  
  props: {  
    className: 'greeting',  
    children: 'Hello, world!'  
  }  
};
```

3.3 JSX의 장점

1) 코드가 간결함

- JSX 사용 시: `<div>Hello, {name}</div>`
- JSX 미사용 시: `React.createElement('div', null, `Hello, ${name}`);`

2) 가독성이 좋음

- 버그를 발견하기 쉬움!

3.3 JSX의 장점

3) Injection Attacks 라 불리는 해킹 방법을 방어

- 보안성이 올라감
- 입력창에 들어가는 문자나 숫자 값 대신 소스코드를 주입하여 그 코드가 실행되도록 만드는 해킹 방법
- ReactDOM은 렌더링하기 전에 JSX내에 포함된 모든 값을 문자열로 바꾸기 때문에 공격을 막을 수 있음

3.4 JSX의 사용법(실습)

- 기본적으로 JS + HTML 코드
 - ... HTML 코드 ... {JS 코드} ... HTML 코드 ...
 - JSX에서는 중괄호를 사용하면 무조건 JS 코드가 들어감
- my-app/src/App.js에서 실습

3.5 JSX 코드 작성해 보기(실습)

- my-app/src/chapter3 실습
 - Book.jsx
 - Library.jsx