



Seoul
Software
ACademy

웹 개발자 부트캠프 과정

SeSAC x CODINGOn

With. 팀 뽀빠

Event Handling

React의 Event

- HTML Event

```
<button onclick="activeEvent();">
  Event Button
</button>
```

- React Event

```
<button onClick={activeEvent}>
  Event Button
</button>
```

React의 Event 주의점


1. React의 이벤트는 소문자가 아닌 **카멜 케이스(camelCase)**를 사용

ex. onclick → onClick


2. JSX를 사용해 이벤트 핸들러를 전달

우측 이미지 처럼 함수 자체를 전달

```
<button onclick={activeEvent()}>
  Event Button
</button>
```



```
<button onClick={activeEvent}>
  Event Button
</button>
```



3. **기본 DOM 요소에만** 이벤트를 설정 가능

React 합성 이벤트

- Synthetic Event (합성 이벤트)
- 모든 브라우저에서 이벤트를 동일하게 처리하기 위한 Wrapper(래퍼) 객체

※ Wrapping이란? 기본 기능을 감싸는 새로운 기능

✧ 리액트가 DOM이 아닌 **VirtualDOM**을 사용하는 것처럼
웹 브라우저의 nativeEvent가 아닌
nativeEvent를 감싼 **SyntheticEvent**를 사용

✧ **SyntheticEvent**
브라우저 이벤트들을 포함하고 있는 **리액트 고유 이벤트 객체**

<https://react.dev/reference/react-dom/components/common#react-event-object>

클래스형 컴포넌트에서의 이벤트

- 클래스형 컴포넌트에서는 **this**를 사용하여 함수를 찾아갈 수 있다.
 - (state와 props와 동일)

```
<button onClick={this.printConsole}>printConsole2</button>
<button onClick={this.handleButton}>handleButton</button>
```

함수에서 this 사용하기

1. bind 사용하기

```
constructor(props) {
  super(props);

  this.eventExample = this.eventExample.bind(this);
}

eventExample() {
  console.log( this )
}
```

2. 화살표 함수 이용하기

```
eventExample2 = () => {
  console.log( this );
}
```

함수에 인자 보내기

1. 화살표 함수 이용하기

```
<button onClick={() => onClickEvent("인자1")}>클릭 이벤트</button>
<button onClick={(e) => onClickEvent2(e, "인자1")}>클릭 이벤트</button>
```

2. bind 사용하기

```
<button onClick={onClickEvent.bind(null, "인자1")}>클릭 이벤트</button>
```


함수형 컴포넌트에서의 이벤트

- 함수에 인자가 없는 경우

```
<button onClick={increase}>더하기</button>
```

- 함수에 인자가 있는 경우

```
<button onClick={() => alertMsg('helloooo')}>alert 출력</button>
<button onClick={(e) => consoleMsg(e, 'helloooo')}>console 출력</button>
```