



이벤트 처리

안 화 수

Event

❖ Event

사용자가 웹 브라우저에서 DOM 요소들과 상호 작용하는 것을 이벤트 (Event) 라고 한다.

Event

❖ Event 종류

- click 이벤트
- change 이벤트
- mouseover 이벤트
- mouseout 이벤트
- key 이벤트 (onKeyUp, onKeyDown, onKeyPress)
- submit 이벤트

Event

❖ Event 처리할때 주의사항

1. 이벤트 이름은 카멜 표기법으로 작성한다.

onclick	---	onClick
onchange	---	onChange
onkeyup	---	onKeyUp

2. DOM 요소에서만 이벤트를 설정 할 수 있다.

div, button, input, form, span 등의 DOM 요소에는 이벤트를 설정할 수 있지만, 우리가 직접 만든 컴포넌트에는 이벤트를 설정 할 수 없다.

`<button onClick={ } > 클릭 </button>` (O)

`<MyComponent onClick={ } />` (X)

Event

❖ 프로젝트 생성

```
npx create-react-app event01
```

Event

❖ event 예제 1 : src/App.js

```
import React from "react";  
import Form01 from "./Form01";  
import TextArea01 from "./TextArea01";  
import Select01 from "./Select01";  
import Reservation from "./Reservation";  
import EventPractice01 from "./EventPractice01";
```

```
function App() {  
  return <Form01/>  
  // return <TextArea01/>  
  // return <Select01/>  
  // return <Reservation/>  
  // return <EventPractice01 />;  
}  
export default App;
```

Event : text

❖ event 예제1 : src/Form01.js (1/2)

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

//function Form01(props){
const Form01 = (props) => {
  const [value, setValue] = useState("");

  const handleChange = (event) => {      // handleChange 메소드
    setValue(event.target.value);
    // setValue(event.target.value.toUpperCase()); //입력한 값을 대문자로 설정
  }
  const handleSubmit = (event) => {      // handleSubmit 메소드
    alert('입력한 이름:' + value);
    event.preventDefault();              // 새로고침 방지
  }
}
```

Event : text

❖ event 예제1 : src/Form01.js (2/2)

```
return (  
  <form onSubmit={handleSubmit}>  
    <label>  
      이름: <input type="text" value={value}  
        onChange={handleChange}/>  
    </label>  
    <button type="submit">제출</button>  
  </form>  
  );  
}
```

```
export default Form01;
```


Event : textarea

❖ event 예제2 : src/TextArea01.js (1/2)

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

//function TextArea01(props){
const TextArea01 = (props) => {
  const [value, setValue] = useState('요청사항을 입력 하세요.');
```



```
  const handleChange = (event) => { // handleChange 메소드
    setValue(event.target.value);    // value값 설정
  }

  const handleSubmit = (event) => { // handleSubmit 메소드
    alert('입력한 요청사항:' + value);
    event.preventDefault();        // 새로그침 방지
  }
}
```

Event : textarea

❖ event 예제2 : src/TextArea01.js (2/2)

```
return (  
  <form onSubmit={handleSubmit}>  
    <label>  
      요청사항: <br/>  
      <textarea cols='40' rows='5' value={value} onChange={handleChange}/>  
    </label>  
    <button type='submit'>제출</button>  
  </form>  
);  
  
}
```

```
export default TextArea01;
```

Event : select

❖ event 예제3 : src/Select01.js (1/2)

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

//function Select01(props){
const Select01=(props)=>{
  const [value, setValue] = useState('grape');

  const handleChange = (event) => { // handleChange 메소드
    setValue(event.target.value);    // value값 설정
  }

  const handleSubmit = (event) => { // handleSubmit 메소드
    alert('선택한 과일:'+ value);
    event.preventDefault();        // 새로고침 방지
  }
}
```

Event : select

❖ event 예제3 : src/Select01.js (2/2)

```
return (  
  <form onSubmit={handleSubmit}>  
    <label>  
      과일을 선택하세요: <br/>  
      <select value={value} onChange={handleChange}>  
        <option value="apple">사과</option>  
        <option value="banana">바나나</option>  
        <option value="grape">포도</option>  
        <option value="watermelon">수박</option>  
      </select>  
    </label>  
    <button type='submit'>제출</button>  
  </form>  
);  
}  
export default Select01;
```

Event

❖ event 예제4 : src/Reservation.js (1/3)

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

```
//function Reservation(props){
```

```
const Reservation=(props)=>{
```

```
  const [haveBreakfast, setHaveBreakfast] = useState(true);
```

```
  const [numberOfGuest, setNumberOfGuest] = useState(2);
```

```
  const handleSubmit = (event) =>{
```

```
    alert('아침식사 여부:'+ haveBreakfast+ ' 방문객 수:'+ numberOfGuest);
```

```
    event.preventDefault();
```

```
  }
```

Event

❖ event 예제4 : src/Reservation.js (2/3)

```
return (  
  <form onSubmit={handleSubmit}>  
    <label>  
      아침식사 여부:  
      <input type='checkbox'  
        checked={haveBreakfast}  
        onChange={(event)=>{  
          setHaveBreakfast(event.target.checked);  
        }}  
      />  
    </label><br/>
```

Event

❖ event 예제4 : src/Reservation.js (3/3)

```
<label>
  방문객 수:
  <input type="number"
    value={numberOfGuest}
    onChange={(event)=>{
      setNumberOfGuest(event.target.value);
    }}
  />
</label>
<button type="submit">제출</button>
</form>
);
}
export default Reservation;
```

Event

❖ event 예제5 : src/EventPractice01.js (1/2)

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

```
//function EventPractice01(){
```

```
  const EventPractice01 = () => {
```

```
    const [username, setUsername] = useState("");
```

```
    const [message, setMessage] = useState("");
```

```
    const onChangeUsername = e => setUsername(e.target.value); //onChangeUsername 메소드
```

```
    const onChangeMessage = e => setMessage(e.target.value); //onChangeMessage 메소드
```

```
    const onClick = () => { // onClick 메소드
```

```
      alert(username + ': ' + message);
```

```
      setUsername("");
```

```
      setMessage("");
```

```
    };
```


Event

❖ event 예제5 : src/EventPractice01.js (2/2)

```
return (  
  <div>  
    <h1>이벤트 연습</h1>  
    <input type="text"  
      name="username"  
      placeholder="유저명"  
      value={username}  
      onChange={onChangeUsername} />  
    <input type="text"  
      name="message"  
      placeholder="아무거나 입력해보세요"  
      value={message}  
      onChange={onChangeMessage} />  
    <button onClick={onClick}>확인</button>  
  </div>  
);  
};  
export default EventPractice01;
```