

Javascript 이벤트

pdf ready

html ready

이벤트 : 마우스 클릭, 키보드 입력 등등 이 발생하면 자바스크립트에 알리는 것이다.

이벤트 종류 (외울 필요 없다)

여기서 **해당** 이란 이벤트 리스너를 등록한 객체를 뜻합니다.

- `onabort` : 해당 이미지 전송이 강제로 중단될 때
- `onblur` : 해당 객체 내의 포커스를 잃었을 때
- `onchange` : 해당 입력 폼의 내용이 변경되고 포커스를 잃을 때
- `onclick` : 마우스 왼쪽 버튼이 해당 객체 위를 클릭 했을 때
- `ondblclick` : 마우스 왼쪽 버튼이 해당 객체 위를 더블 클릭했을 때
- `onerror` : 해당 이미지의 로딩이 실패했을 때
- `onfocus` : 해당 폼이 포커스를 받았을 때
- `onkeydown` : 해당 객체가 포커스를 받았을 때 키보드를 누르면 발생
- `onkeypress` : 해당 객체가 포커스를 받았을 때 키보드를 눌렀다 떼면 발생
- `onkeyup` : 해당 객체가 포커스를 받았을 때 키보드를 떼면 발생
- `onmousemove` : 해당 객체 위에서 마우스가 움직일 때 발생

포커스를 안받아도 발생된다.

- `onmouseout` : 마우스가 해당 객체를 나갈 때
- `onmouseover` : 마우스가 해당 객체에 올라갔을 때

`onmouseout` 과 `onmouseover` 은 반대 역할이다.

해당 객체 위에 마우스를 올렸을 때 `onmouseover` 이 발생하고

해당 객체 위에 마우스를 올린 상태에서 마우스가 나가면 `onmouseout` 이 발생한다.

둘다 한번씩 발생한다.

- `onmousedown` : 마우스가 해당 객체 위에서 마우스의 왼쪽, 오른쪽, 휠 버튼을 누를 때 발생한다.

`onclick` 과 `onmousedown` 의 차이

- `onclick` 은 마우스 왼쪽 버튼이 클릭했을 때이고
 - `onmousedown` 은 마우스의 세 버튼중 하나라도 눌렀을 때 발생한다.
- `onmouseup` : `onmousedown` 뒤에 마우스 버튼을 떼 때 발생
- `mouseup`, `mousedown` 이벤트는 `keyup`, `keydown` 과 동일하게 동작한다. 단지 마우스 나 키보드냐의 차이.
- `onresize` : 사용자나 스크립트로 인해 해당 객체의 크기가 변경됐을 때 작동.
테스트 해 본 결과 `body` 태그에서만 작동이 된다. 왜인지는 모르겠다. 추가 바람.
 - `onselect` : 해당 입력 폼의 내용이 선택됐을 때
 - `onreset` : form 태그 안의 입력 폼이 `<input type=reset >` 이나 `form.reset()` 으로 인해 리셋됐을 때.

```
<form onreset="console.log('reset')">
  이름 : <input type="text">
  <input type="reset" value="리셋">
</form>
```

- `onsubmit` : form 태그의 `<input type="submit">` 이나 `form.submit()` 으로 인해 입력 폼이 제출됐을 때

```
<form onsubmit="console.log('submit')">
  이름 : <input type="text">
  <input type="submit" value="제출">
</form>
```

- `<body onload` : 문서의 로딩이 다 되었을 때

역시나 `body` 태그에만 작동이 된다.

`window.onload` 와 이 `onload` 는 동일하게 동작한다고 한다.

- `onunload` : 문서가 언로드 될 때

우리가 웹사이트를 사용하다 보면 사이트를 종료하기 전에 경고문이 뜰 때가 있다.

방법은 `onunload` 에 질문 창을 넣은 것이다.

이벤트 리스너 생성 방법

1. HTML 태그 내에 작성

```
<element onclick="doClick(event)"></element> <!--html 태그 내에서 이벤트 객체는 ever
```

2. DOM 객체의 프로퍼티 내에 작성

```
document.getElementById("password").onclick = doclick;
```

3. addEventListener 메소드

```
document.body.addEventListener('click', doclick);
```

이벤트 객체 생성 및 소멸 단계

1. 이벤트 발생

사용자가 마우스를 클릭함

2. 이벤트 정보를 담은 이벤트 객체를 생성

해당 마우스 클릭의 정보를 담은 이벤트 객체를 생성

```
evt = {...};
```

3. 이벤트 객체를 이벤트 리스너에 전달

onclick(evt) evt는 마우스 클릭의 정보

4. 이벤트 처리

```
function doclick(e)
{
    console.log('클릭했습니당 정보 : ' , e);
}
```

5. 이벤트 객체 소멸

```
document.removeEventListener('click', doclick);
```

e.preventDefault

기본 이벤트를 막는다.

```
<form action="/hello" method="get" id="loginForm">
  <input type="text" name="id">
  <input type="submit" value="로그인">
</form>
```

```
function login(e) {
  e.preventDefault(); // 자동으로 redirect 되는 것을 막음
  console.log("폼 제출 막기");
}
document.getElementById("loginForm").onsubmit = login;
```

이렇게 작성하면 form이 제출되는 것을 막는다.