

# 이벤트

# ▶ 이벤트

## ✓ 이벤트 활용

이벤트 속성과 이벤트 핸들러(함수)를 연동하여  
이벤트 발생 시 특정 기능을 하도록 하는 것

## ✓ 이벤트 설정 방법

- 고전 이벤트 모델
- 인라인 이벤트 모델
- 표준 이벤트 모델
- 마이크로소프트 인터넷 익스플로러 이벤트 모델

# ▶ 이벤트 설정 방법

## ✓ 고전 이벤트 모델

요소 객체가 갖고 있는 이벤트 속성에 이벤트 핸들러를 연결하는 방법으로 이벤트 객체를 제거할 땐 속성 값에 null을 넣어주면 됨  
이벤트 발생 객체는 핸들러 내부에서 this로 표현(스타일 변경 가능)  
매개 변수로 이벤트 정보 전달(e, window.event) \* 이벤트 객체 전달

ex. 클릭 시 이벤트 설정

```
var h = document.getElementById('id명');  
h.onclick = function(  
    수행 기능 설정;  
    h(this).onclick = null; // 한 번만 실행  
);
```

# ▶ 이벤트 설정 방법

## ✓ 인라인 이벤트 모델

요소 내부에 이벤트를 작성하는 방식으로  
인라인 방식은 <script>태그에 있는 함수를 호출하는 방식 선호

ex. 클릭 시 이벤트 설정

```
<h1 onclick='처리로직'> </h1>
```

또는

```
<h1 onclick='스크립트 내 함수 호출'> </h1>
```

# ▶ 이벤트 설정 방법

## ✓ 표준 이벤트 모델

w3에서 공식적으로 지정한 이벤트 모델  
한 번에 여러 개 이벤트 핸들러 설정 가능  
this키워드가 이벤트 발생 객체를 의미함

ex. 클릭 시 이벤트 설정

```
var h = document.getElementById('id명');  
h.addEventListener('click', function(){수행 기능 설정;});
```

## ✓ 메소드

addEventListener(이벤트이름, 핸들러, 확장)	확장 : 버블링/캡처링
removeEventListener(이벤트이름, 핸들러)	이벤트 삭제

# ▶ 이벤트 설정 방법

## ✓ 익스플로러 이벤트 모델

익스플로러 브라우저 적용 모델

한 번에 여러 개 이벤트 핸들러 설정 가능

this키워드는 이벤트 발생객체(srcElement)가 아닌 window객체 의미

ex. 클릭 시 이벤트 설정

```
var h = document.getElementById('id명');
```

```
h.attachEvent('onclick', function(){수행기능 설정;});
```

## ✓ 메소드

attachEvent(이벤트이름, 핸들러)	익스플로러에만 존재하는 메소드
detachEvent(이벤트이름, 핸들러)	이벤트 삭제

# ▶ 기본 이벤트 제거/유효성 검사

- 기본 이벤트 : 태그 중 이벤트 핸들러를 기본적으로 가지고 있는 것
  - \* <a>, <input type="submit"> 입력 양식에서 많이 사용
- 유효성 검사 : 사용자가 입력한 데이터가 양식에 맞는지 검사 ex. 비밀번호 확인

## ✓ 기본 이벤트 제거

onsubmit이벤트 속성에 이벤트 핸들러 연결할 때 return false;함

## ✓ 유효성 검사

이벤트 핸들러에 데이터 비교 후 맞지 않으면 false 리턴

\* 정규식 문서 참고

# ▶ 이벤트 전달 및 차단

## ✓ 이벤트 전달

- 버블링 방식 : 자식노드에서 부모노드로 올라가면서 이벤트 실행
  - 캡처링 방식 : 부모노드에서 자식노드로 내려가면서 이벤트 실행
- \* 익스플로러는 지원하지 않음

## ✓ 이벤트 차단

이벤트가 전달되어 모든 이벤트가 발생하는 것을 차단

이벤트객체.stopPropagation()	익스플로러 제외한 브라우저
window.event.cancelBubble=true;	익스플로러