

이벤트

강사 : 강병준

❖ jQuery로 이벤트 다루기

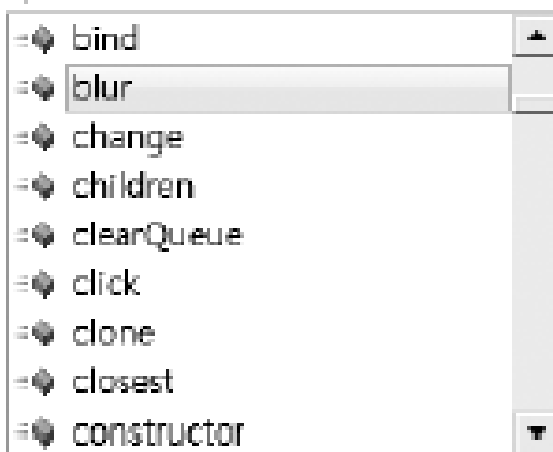
- 보조 기능을 사용해 어떠한 메서드가 이벤트와 관련되는지 살펴보기

`$(document).ready()` 메서드

```
$(document).ready(function (event) {  
  
});
```

jQuery의 이벤트 메서드

`$(document).d`



jQuery에서 지원하는 이벤트

이벤트 명	발생 위치	정 의
BLUR	FOCUS가 가능한 엘리먼트	엘리먼트가 포커스를 잃을 때 발생 (포커스 : 커서가 객체 안에 위치하거나, TAB 키 등으로 해당 엘리먼트가 선택된 경우)
FOCUS	FOCUS가 가능한 엘리먼트	엘리먼트가 포커스를 얻을 때 발생
LOAD	모든 엘리먼트	엘리먼트와 자신과 모든 서브엘리먼트들이 로딩을 마쳤을 때 발생
RESIZE	WINDOW 엘리먼트	브라우저의 크기가 변경되었을 때 WINDOW 객체에서 발생
CLICK	모든 엘리먼트	엘리먼트 위에 마우스 포인터를 위치시키고, 마우스 버튼을 누르고 땀을 때 발생

jQuery에서 지원하는 이벤트

이벤트 명	발생 위치	정 의
DBLCLICK	모든 엘리먼트	엘리먼트가 더블 클릭 되었을 때 발생
MOUSEDOWN	모든 엘리먼트	엘리먼트 위에 마우스 포인터가 위치되어있고, 마우스 버튼을 눌렀을 때 발생
MOUSEUP	모든 엘리먼트	엘리먼트 위에 마우스 포인터가 위치되어있고, 마우스 버튼을 놓을 때 발생
MOUSEMOVE	모든 엘리먼트	마우스 포인터가 엘리먼트 내부에서 움직일 때 발생
MOUSEOVER	모든 엘리먼트	마우스 포인터가 엘리먼트 안으로 이동할 때 발생

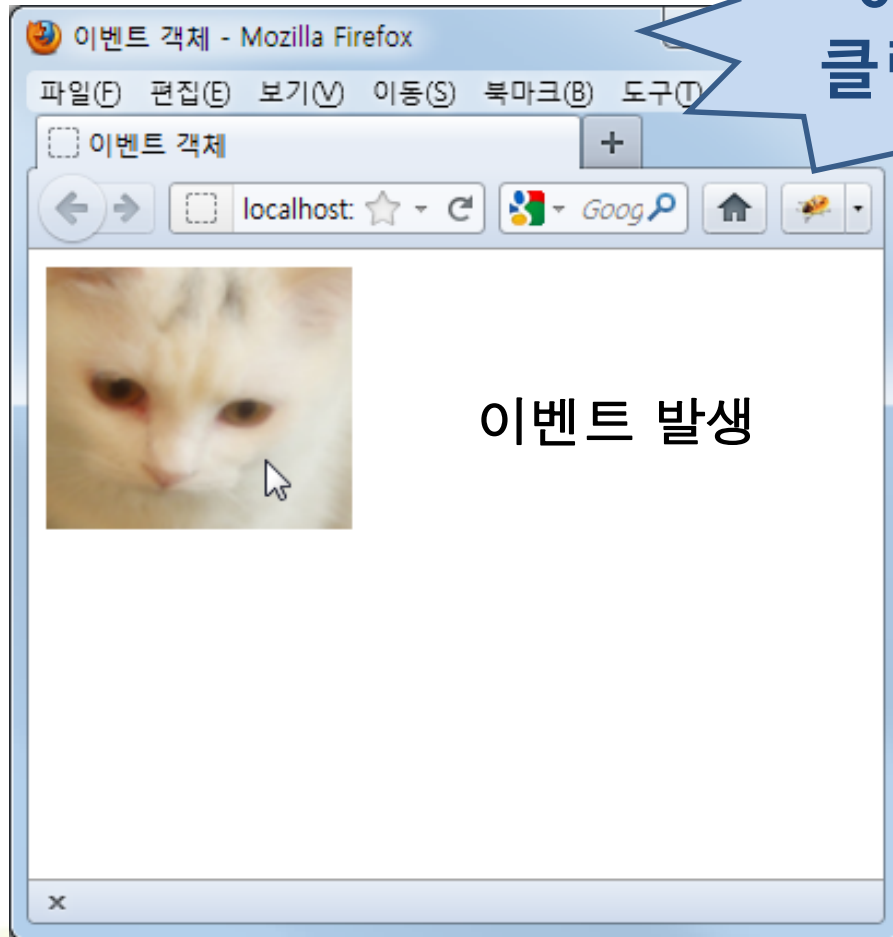
jQuery에서 지원하는 이벤트

이벤트 명	발생 위치	정 의
MOUSEOUT	모든 엘리먼트	마우스 포인터가 엘리먼트에서 벗어날 때 발생
SELECT	<INPUT TYPE="TEXT"> <TEXTAREA>	내부의 텍스트들을 선택했을 때(영역이 반전되었을 때) 발생
SUBMIT	<INPUT TYPE="SUBMIT">	FORM 값을 넘기려(SUBMIT) 할 때
KEYDOWN	FOCUS가 가능한 엘리먼트	키보드를 누를 때 발생, 키를 누르고 있어도 1회만 발생
KEYPRESS	FOCUS가 가능한 엘리먼트	키보드를 누를 때 발생, 키를 누르고 있으면 연속적으로 발생

jQuery에서 지원하는 이벤트

이벤트 명	발생 위치	정 의
KEYUP	FOCUS가 가능한 엘리먼트	키보드에서 손을 땄 때 발생
ERROR	로딩이 일어나는 엘리먼트	이미지 등이 정상적으로 로딩되지 않았을 때 발생
SELECT	<INPUT TYPE="TEXT"> <TEXTAREA>	내부의 텍스트들을 선택했을 때(영역이 반전되었을 때) 발생

이벤트 처리



이미지가
클릭되었다.

이벤트 핸들러

이벤트 연결

❖ jQuery 사용해 이벤트 연결

- 가장 기본적인 방법은 bind() 메서드 사용
- bind() 메서드는 다음과 같은 두 가지 형태로 사용
 - 1 \$(selector).bind(eventName, function(event) { })
 - 2 \$(selector).bind(object)

```
<body>  
  <h1>Header-0</h1>  
  <h1>Header-1</h1>  
  <h1>Header-2</h1>  
</body>
```


이벤트 연결

❖ **\$(selector).bind(eventName, function(event) { })**

- bind() 메서드의 첫 번째 매개 변수에 이벤트 이름 입력
- 두 번째 매개 변수에 이벤트 핸들러 입력
- 이벤트 핸들러 안에서 this 키워드는 이벤트 발생한 객체 의미
 - h1 태그를 click 이벤트에 연결
 - 이벤트 발생 시 이벤트 발생 객체에 '+' 글자 추가 bind01.html

```
<script src="http://code.jquery.com/jquery-1.7.js"></script>
```

```
<script>
```

```
$(document).ready(function () {
```

```
    // 이벤트를 연결합니다.
```

```
    $('h1').bind('click', function () {
```

```
        $(this).html(function (index, html) {
```

```
            return html + '+';
```

```
        });
```

```
    });
```

```
});
```

```
</script>
```

이벤트 연결

❖ 2 \$(selector).bind(object) bind02.html

- bind() 메서드의 매개 변수에 객체 넣어줌
- 속성 이름과 속성값에 이벤트 이름과 이벤트 핸들러 넣음
 - 마우스가 진입하면 reverse 클래스 속성 추가
 - 마우스가 빠져나가면 reverse 클래스 속성 제거

```
<script>
$(document).ready(function () {
    // 이벤트를 연결합니다.
    $('h1').bind('click', function () {
        $(this).html(function (index, html) {
            return html + '+';
        });
    });

    // 이벤트를 연결합니다.
    $('h1').bind({
        mouseenter: function () { $(this).addClass('reverse'); },
        mouseleave: function () { $(this).removeClass('reverse'); }
    });
});
</script>
```

이벤트 연결

❖ Style 태그 입력

```
<!DOCTYPE html>
<html>
<head>
  <style>
    .reverse {
      background: Black;
      color: White;
    }
  </style>
```

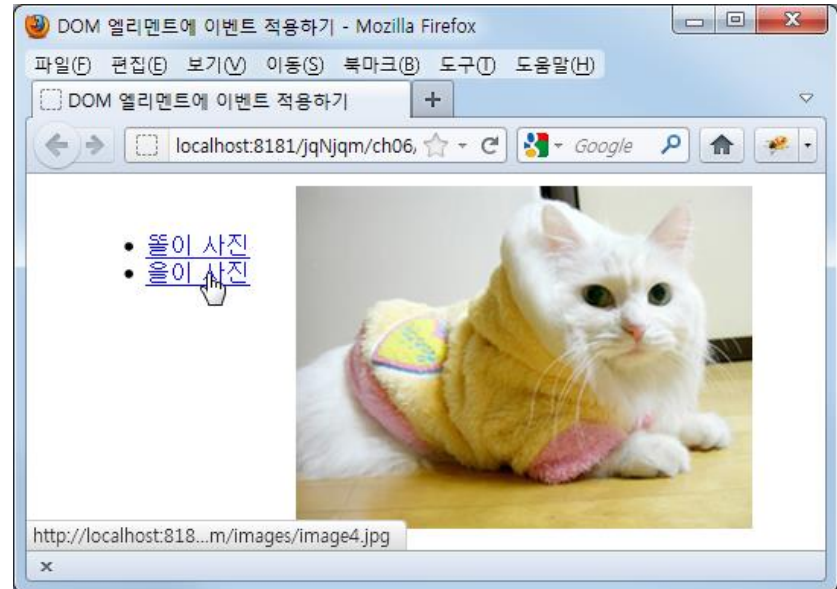
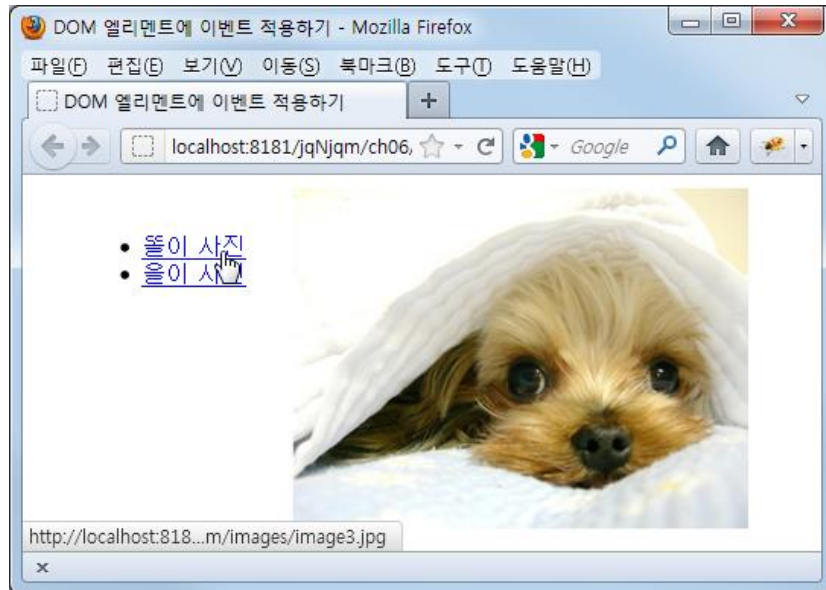
이벤트 연결

Header-0+++

Header-1+++++

Header-2+

이벤트 등록하기 -1



이벤트 등록하기 -2

```
$('#a').bind('click', function(){  
    $("img").attr("src", $(this).attr("href"));  
    return false;  
});
```

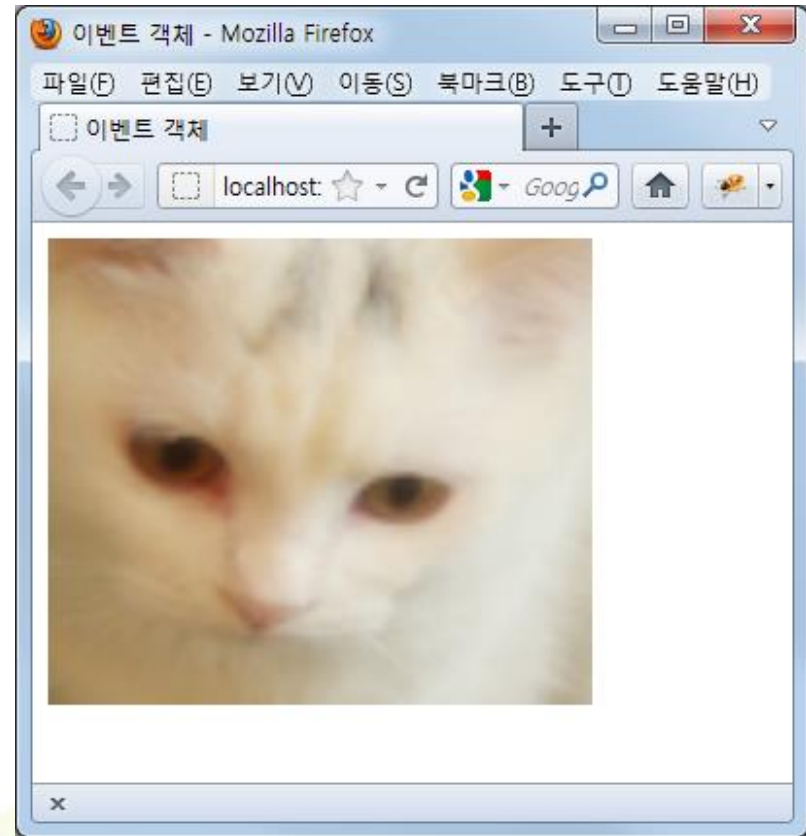
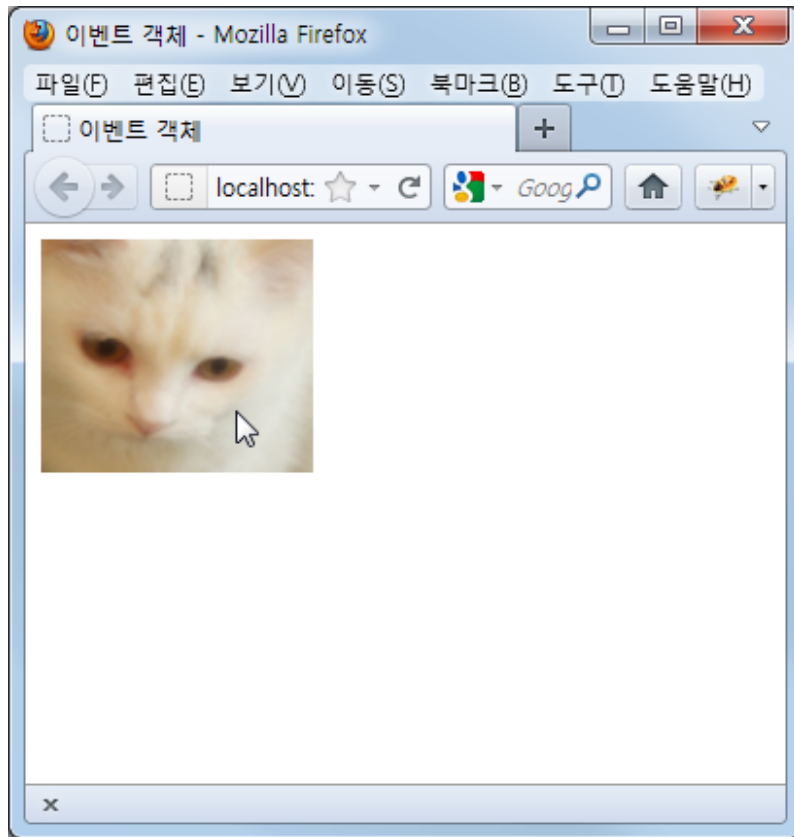
■ bind(‘이벤트타입이름’ , 이벤트핸들러)

- 이 메소드로 해당 엘리먼트에 원하는 이벤트를 적용할 수 있다. 적용 가능한 이벤트에는 click, mousedown, focus, resize 등 여러 가지가 있다. 인자에 기술한 함수는 이벤트가 발생했을 시 자동 실행된다.

```
$('#a').click(function(){  
    $("img").attr("src", $(this).attr("href"));  
    return false;  
});
```

```
<!DOCTYPE html>
<html>
<head>
<style type="text/css">
ul { float:left; margin:30px}
</style>
<meta charset="UTF-8">
<title>DOM 엘리먼트에 이벤트 적용하기</title>
<script type="text/javascript" src="../js/jquery.js"></script>
<script type="text/javascript">
    $(function(){
        $('a').bind('click',function(){
            $("img").attr("src", $(this).attr("href"));
            return false;
        });
    });
</script>
</head>
<body>
    <ul>
        <li> <a href="../images/image3.jpg"> 딸이 사진 </a> </li>
        <li> <a href="../images/image4.jpg"> 을이 사진 </a> </li>
    </ul>
    <img>
</body>
</html>
```

이벤트 객체-event.target



이벤트 객체-event.target

```
$('#img').bind('click', function(event){  
    var $target = $(event.target);  
    $target.width($target.width()*2);  
    $target.height($target.height()*2);  
});
```

■ event 인스턴스

- 동작함수의 인자인 event 인스턴스는 함수가 실행될 때 브라우저로부터 전달한 값을 갖는다.

■ event 인스턴스 프로퍼티

- data : 핸들러를 할당할 때 bind() 메소드의 두번째 매개변수로 전달된 값
- target : 이벤트가 발생한 대상 엘리먼트


```
<!DOCTYPE html><html><head>
<meta charset="UTF-8">
<title>이벤트 객체</title>
<style type="text/css">
img{
    width: 140px; height: 120px;
}
</style>
<script type="text/javascript" src="../js/jquery.js"></script>
<script type="text/javascript">
    (function(){
        $('img').bind('click', function() {
            $(this).width($(this).width()*1.5);
            //$(this).height($(this).height()*1.5);
        });
    });
</script>
</head>
<body>
    
</body>
</html>
```

일회성 이벤트 만들기

one('이벤트 종류', data, 동작함수)

```
$('#img').one('click', function(event){  
    var $target = $(event.target);  
    $target.width($target.width()*2);  
    $target.height($target.height()*2);  
});
```

■ one()

- .bind()와 동일하지만 이벤트가 한번 실행된 후 .unbind()된다.

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>일회성 이벤트 만들기</title>
<style type="text/css">
img{
    width: 140px; height: 120px;
}
</style>
<script type="text/javascript" src="../js/jquery.js"></script>
<script type="text/javascript">
    $(function(){
        $('img').one('click', function(event){
            $(this).width($(this).width()*1.5);
            // $(this).height($(this).height()*1.5);
        });
    });
</script>
</head>
<body>
    
</body>
</html>
```

간단한 이벤트 연결

❖ jQuery 단축 이벤트 메서드

■ 사용법

- `$(selector).method(function(event) { });`

jQuery 단축 이벤트 메서드

blur	focus	focusin	focusout	load
resize	scroll	unload	click	dblclick
mousedown	mouseup	mousemove	mouseover	mouseout
mouseenter	mouseleave	change	select	submit
keydown	keypress	keyup	error	ready

jQuery의 메서드

메서드 이름	설명
<code>toggle()</code>	click 이벤트를 여러 이벤트 핸들러를 번갈아가며 실행할 수 있게 연결합니다.
<code>hover()</code>	mouseenter 이벤트와 mouseleave 이벤트를 동시에 연결합니다.

간단한 이벤트 연결

❖ jQuery 단축 이벤트 메서드 구현

- HTML 페이지 구성

```
<!DOCTYPE html><html><head>
  <style>
    .reverse {
      background: Black;
      color: White;
    }
  </style>
  <script src="http://code.jquery.com/jquery-1.7.js"></script>
  <script>
    $(document).ready(function () {

      });
  </script>
</head>
<body>
  <h1>Header-0</h1>
  <h1>Header-1</h1>
  <h1>Header-2</h1>
</body></html>
```

간단한 이벤트 연결

❖ hover() 메서드

- 두 개의 함수를 매개 변수로 넣어줌
 - 첫 번째 매개 변수에 넣는 함수는 mouseenter 이벤트 핸들러
 - 두 번째 매개 변수에 넣는 함수는 mouseleave 이벤트 핸들러

```
<style>
.reverse {
    background: Black;
    color: White;
}
</style>
<script src="http://code.jquery.com/jquery-1.7.js"></script>
<script>
$(document).ready(function () {
    // 이벤트를 연결합니다.
    $('h1').hover(function () {
        $(this).addClass('reverse');
    }, function () {
        $(this).removeClass('reverse');
    });
});
</script>
```

이벤트 연결 제거

❖ 이벤트 제거 시 unbind() 메서드 사용

■ unbind() 메서드의 사용 형태

- 1 \$(selector).unbind() - 해당 문서 객체와 관련된 모든 이벤트 제거
- 2 \$(selector).unbind(eventName) - 특정 이벤트와 관련된 모든 이벤트 제거
- 3 \$(selector).unbind(eventName, function) - 특정 이벤트 핸들러 제거

■ body 태그 구성

```
<body>  
  <h1>Header-0</h1>  
  <h1>Header-1</h1>  
  <h1>Header-2</h1>  
</body>
```

필요 없는 이벤트 제거하기

```
$('#img').bind('click', function(event){  
    $(this).width ($(this).width()*2);  
    $(this).height ($(this).height()*2);  
    $(this).unbind();  
});
```

- **unbind(‘이벤트 종류’ , data, 동작함수)**
 - 지정된 이벤트 핸들러를 삭제한다.

이벤트 연결 제거

- h1 태그에 click 이벤트 연결하고 이벤트 발생 알람
- 이후에 unbind() 메서드 사용해 이벤트 제거
- 같은 기능 하는 one() 메서드 존재 : 이벤트를 한번만 연결

```
<script src="http://code.jquery.com/jquery-1.7.js"></script>
<script>
    $(document).ready(function () {
        // 이벤트를 연결합니다.
        $('h1').click(function () {
            // 출력합니다.
            $(this).html('CLICK');
            alert('이벤트가 발생했습니다. ');
            // 이벤트를 제거합니다.
            $(this).unbind();
        });
    });
</script>
```

매개 변수 context

❖ 매개 변수 context

- jQuery 메서드는 매개 변수 두 개 입력 가능
- selector 매개 변수 이외에 context 매개 변수 가능
- 매개 변수 context = selector 적용하는 범위 한정

jQuery 메서드의 형태

`$({`

`$(selector, context)`

매개 변수 context

❖ 매개 변수 context 예제

- 특정 부분에 선택자 적용하고 싶을 때 사용하는 매개 변수
- Body 태그 구성

```
<body>
  <div>
    <h1>Header 1</h1>
    <p>Paragraph</p>
  </div>
  <div>
    <h1>Header 2</h1>
    <p>Paragraph</p>
  </div>
  <div>
    <h1>Header 3</h1>
    <p>Paragraph</p>
  </div>
</body>
</html>
```

매개 변수 context

❖ 매개 변수 context 예제

■ Style 태그 구성

```
<!DOCTYPE html>
<html>
<head>
  <style>
    * { margin:0px; padding:0px }
    div
    {
      margin:5px; padding:3px;
      border-width:3px;
      border-style:solid;
      border-color:Black;
      border-radius:10px;
    }
  </style>
```

매개 변수 context

❖ 매개 변수 context 예제

- Context 객체 사용

```
<script src="http://code.jquery.com/jquery-1.7.js"></script>
<script>
    $(document).ready(function () {
        $('div').click(function () {
            var header = $('h1', this).text();
            var paragraph = $('p', this).text();
            alert(header + '\n' + paragraph);
        });
    });
</script></head><body>
    <div><h1>Header 1</h1> <p>Paragraph</p></div>
    <div><h1>Header 2</h1> <p>Paragraph</p></div>
    <div><h1>Header 3</h1> <p>Paragraph</p></div>
</body>
</html>
```

이벤트 객체

❖ jQuery가 이벤트 객체 정형화

- jQuery의 이벤트 객체는 모든 브라우저가 같은 방법으로 사용
- 같은 속성을 가짐

❖ 자주 사용하는 jQuery 이벤트 객체의 속성

jQuery 이벤트 객체의 속성

이벤트 객체 속성	설명
event.pageX	브라우저의 화면을 기준으로 한 마우스의 X 좌표 위치
event.pageY	브라우저의 화면을 기준으로 한 마우스의 Y 좌표 위치
event.preventDefault()	기본 이벤트를 제거합니다.
event.stopPropagation()	이벤트 전달을 제거합니다.

이벤트 강제 발생

❖ 이벤트 강제 발생

- trigger() 메서드 사용 예 ; 이벤트를 강제로 발생시킴
 - 1 \$(selector).trigger(eventName)
 - 2 \$(selector).trigger(eventName, data)
 - 2번 형태의 매개 변수 data는 일반적으로 배열
- 1번 형태 예제 Body 태그 구성

```
<body>  
  <h1>Start: </h1>  
  <h1>Start: </h1>  
</body>
```

이벤트 강제 발생

❖ trigger() 메서드

- h1 태그에 click 이벤트 연결
 - 클릭하면 뒤에 별 추가
 - setInterval() 함수 사용

– **마지막에 위치한 h1 태그는 자동으로 1초마다 클릭 이벤트 실행**

```
<script src="http://code.jquery.com/jquery-1.7.js"></script>
```

```
<script>
```

```
$(document).ready(function () {
```

```
    // 이벤트를 연결합니다.
```

```
    $('h1').click(function () {
```

```
        $(this).html(function (index, html) {
```

```
            return html + '★';
```

```
        });
```

```
    });
```

```
    // 1초마다 함수를 실행합니다.
```

```
    setInterval(function () {
```

```
        $('h1').last().trigger('click');
```

```
    }, 1000);
```

```
});
```

```
</script>
```

실행 결과

Start: ★★★★★★★★★★

Start: ★★★★★

이벤트 강제 발생

❖ 간단한 방식으로 이벤트 강제 실행

```
<script src="http://code.jquery.com/jquery-1.7.js"></script>
<script>
    $(document).ready(function () {
        // 이벤트를 연결합니다.
        $('h1').click(function () {
            $(this).html(function (index, html) {
                return html + '★';
            });
        });

        // 1초마다 함수를 실행합니다.
        setInterval(function () {
            $('h1').last().click();
        }, 1000);
    });
</script>
```

이벤트 강제 발생

❖ trigger() 메서드의 2번 형태 실행

- 2 \$(selector).trigger(eventName, data)

```
<!DOCTYPE html>
<html>
<head>
  <script src="http://code.jquery.com/jquery-1.7.js"></script>
  <script>
    $(document).ready(function () {
      // 이벤트를 연결합니다.
      $('h1').click(function (event, data1, data2) {
        alert(data1 + ' : ' + data2);
      });
      // 이벤트를 강제로 발생시킵니다.
      $('h1').eq(1).trigger('click', [273, 52]);
    });
  </script>
</head>
<body>
  <h1>TRIGGER</h1> <h1>TRIGGER</h1>
</body>
</html>
```

기본 이벤트와 이벤트 전달

❖ 기본 이벤트 제거, 이벤트 전달을 막는 메서드

- preventDefault() 기본이벤트 제거
- stopPropagation() 이벤트 전달을 제거

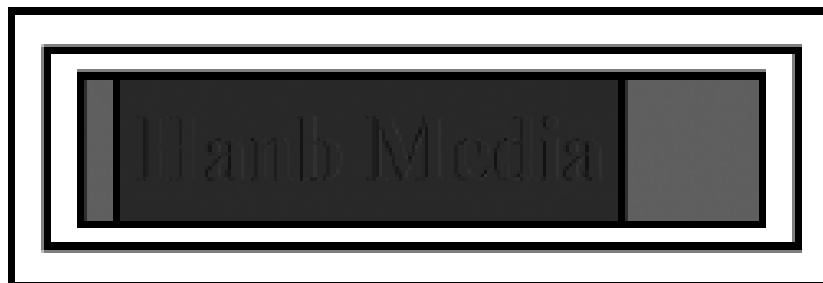
```
<!DOCTYPE html><html><head>
  <style>
    * {
      margin:5px; padding:5px;
      border:3px solid Black;
    }
  </style>
  <script src="http://code.jquery.com/jquery-1.7.js"></script>
  <script>
    $(document).ready(function () {
    });
  </script>
</head>
<body>
  <h1>
    <a href="http://www.choongang.co.kr">중앙정보처리학원</a>
  </h1>
</body></html>
```

기본 이벤트와 이벤트 전달

- preventDefault() 메서드
 - H1 태그 눌렀을 때 자동 실행되는 a 태그 기본 이벤트 제거

```
<script>
$(document).ready(function () {
    $('a').click(function (event) {
        $(this).css('background-color', 'Blue');
        event.preventDefault();
    });
    $('h1').click(function () {
        $(this).css('background-color', 'Red');
    });
});
</script>
```

anchor 태그를 클릭했을 경우의 실행 결과



기본 이벤트와 이벤트 전달

- 이벤트 객체의 stopPropagation() 메서드 사용

```
<style>
  * {
    margin:5px; padding:5px;
    border:3px solid Black;
  }
</style>
<script src="http://code.jquery.com/jquery-1.7.js"></script>
<script>
  $(document).ready(function () {
    $('a').click(function (event) {
      $(this).css('background-color', 'Blue');
      event.stopPropagation();
      event.preventDefault();
    });

    $('h1').click(function () {
      $(this).css('background-color', 'Red');
    });
  });
</script>
```

anchor 태그를 클릭했을 경우의 실행 결과



기본 이벤트와 이벤트 전달

- stopPropagation() 메서드와 preventDefault() 메서드 함께 사용
- jQuery는 return false 사용
 - 이 두 가지 함께 적용하는 것으로 인식

```
<script src="http://code.jquery.com/jquery-1.7.js"></script>
<script>
    $(document).ready(function () {
        $('a').click(function (event) {
            $(this).css('background-color', 'Blue');
            return false;
        });

        $('h1').click(function () {
            $(this).css('background-color', 'Red');
        });
    });
</script>
```

마우스 이벤트

❖ 마우스 이벤트

마우스 이벤트

이벤트 이름	설명
click	마우스를 클릭할 때 발생합니다.
dblclick	마우스를 더블클릭할 때 발생합니다.
mousedown	마우스 버튼을 누를 때 발생합니다.
mouseup	마우스 버튼을 뗄 때 발생합니다.
mouseenter	마우스가 요소의 경계 외부에서 내부로 이동할 때 발생합니다.
mouseleave	마우스가 요소의 경계 내부에서 외부로 이동할 때 발생합니다.
mousemove	마우스를 움직일 때 발생합니다.
mouseout	마우스가 요소를 벗어날 때 발생합니다.
mouseover	마우스를 요소 안에 들어올 때 발생합니다.

마우스 이벤트

- div 태그를 두 개 중첩해서 만든 후
- 외부 div 태그에 mouseover 이벤트와 mouseenter 이벤트 모두 연결

```
<!DOCTYPE html><html><head>
```

```
<style>
```

```
.outer { width:200px; height:200px; background:Orange; padding:50px; margin:10px;}
```

```
.inner { width:100%; height:100%; background:Pink; }
```

```
</style>
```

```
<script src="http://code.jquery.com/jquery-1.7.js"></script>
```

```
<script>
```

```
$(document).ready(function () {
```

```
  $('outer').mouseover(function () {
```

```
    $('body').append('<h1>MOUSEOVER</h1>');
```

```
  }).mouseenter(function () {
```

```
    $('body').append('<h1>MOUSEENTER</h1>');
```

```
  });
```

```
});
```

```
</script></head>
```

```
<body>
```

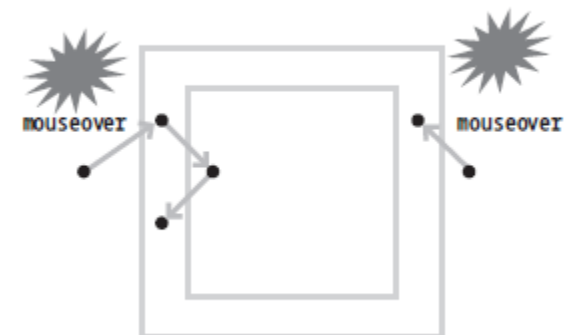
```
<div class="outer">
```

```
<div class="inner"></div>
```

```
</div>
```

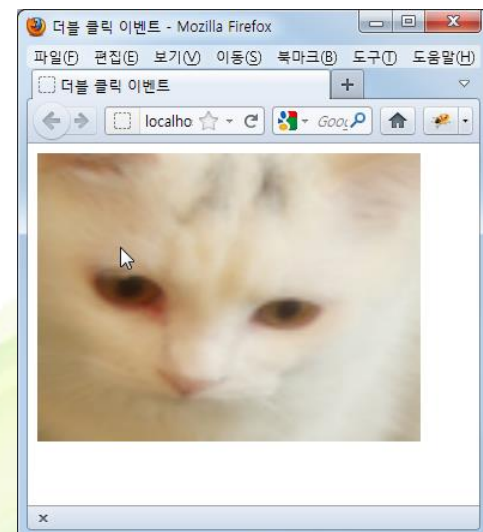
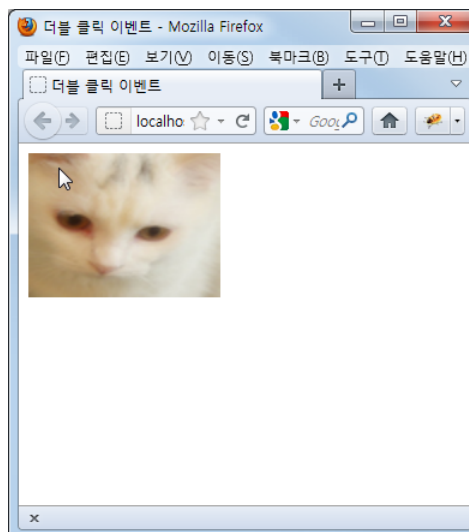
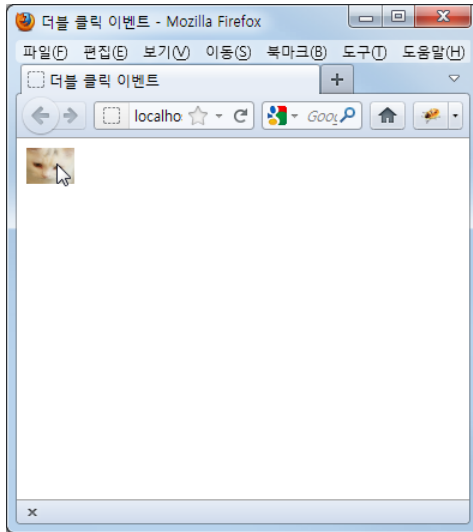
```
</body></html>
```

mouseover와 mouseenter 이벤트의 차이



그림을 클릭할 때마다 그림 커지는 예제

```
$('img').dblclick(function(event){  
    var $target = $(event.target);  
    $target.width($target.width()+20);  
    $target.height($target.height()+20);  
});
```



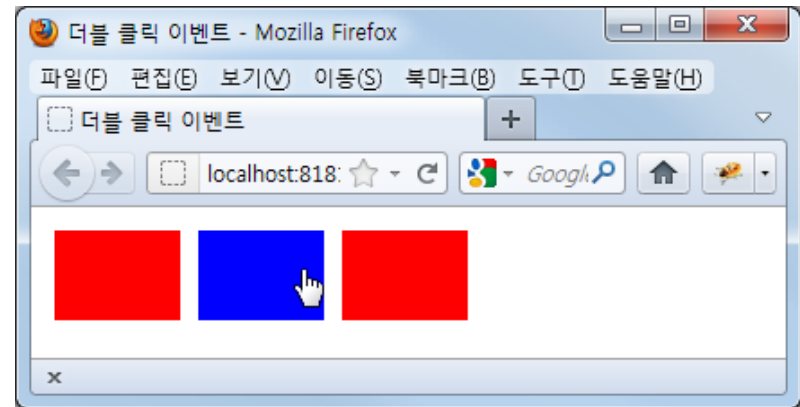
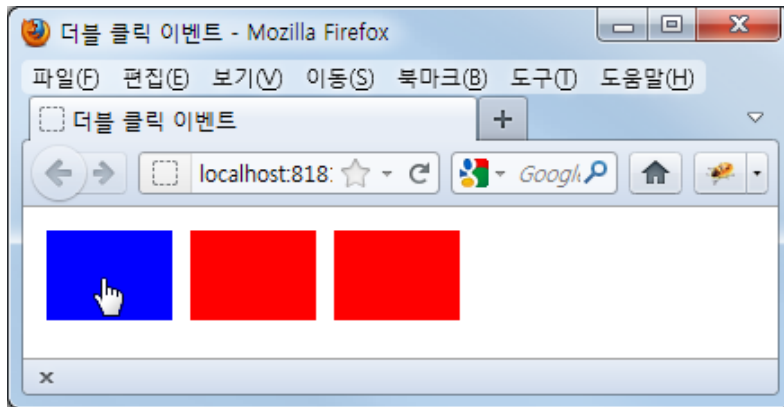
■ **dblclick()**

- 더블 클릭할 때 발생하는 click 이벤트를 붙인다.

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>더블 클릭 이벤트</title>
<style type="text/css">
img{
    width: 40px; height: 30px;
}
</style>
<script type="text/javascript" src="../js/jquery.js"></script>
<script type="text/javascript">
    $(function(){
        $('img').dblclick(function(event){
            // $('img').bind('dblclick', function(event){
                $(this).width($(this).width()+20);
                // $(this).height($(this).height()+20);
            });
    });
</script>
</head>
<body>
    
</body>
</html>
```

마우스를 누르면 파란색, 놓으면 빨간색으로 배경색 바꾸기

```
$('#div').mousedown(function(event){  
    $(event.target).css('backgroundColor', 'blue');  
});  
$('#div').mouseup(function(event){  
    $(event.target).css('backgroundColor', 'red');  
});
```



- **.mousedown()**
 - 마우스 버튼을 내릴 때 발생하는 mousedown 이벤트를 붙인다.
- **mouseup()**
 - 마우스 버튼을 올릴 때 발생하는 mouseup 이벤트를 붙인다.

```
<!DOCTYPE html><html><head><meta charset="UTF-8">
<title>마우스를 누르면 파란색, 놓으면 빨간색으로 배경색 바꾸기</title>
<style type="text/css">
div{ width: 70px; height: 50px; float:left;
    background-color: red; cursor: pointer;
    margin: 5px
}
</style>
<script type="text/javascript" src="../js/jquery.js"></script>
<script type="text/javascript">
    $(function(){
        $('div').mousedown(function() {
            $(this).css('background','blue');
        });
        $('div').mouseup(function() {
            $(this).css('background','red');
        });
    });
</script></head><body>
    <div></div> <div></div> <div></div>
</body>
</html>
```

마우스가 움직인 곳의 위치 알아내기

```
$('#div').mousemove(function(event){  
    $('#p:first').html(' 페이지를 기준으로 한 x 좌표 : '  
        + event.clientX  
        + '<br/> 페이지를 기준으로 한 y 좌표 : '  
        + event.clientY + '<br/>');  
    $('#p:last').html(' 화면을 기준으로 한 x 좌표 : '  
        + event.screenX  
        + '<br/> 화면을 기준으로 한 y 좌표 : '  
        + event.screenY + '<br/>');  
});
```

■ **mousemove(function)**

- 마우스 포인터가 올라갈 때 발생하는 mouseover 이벤트를 붙인다.

```
<!DOCTYPE html><html><head><meta charset="UTF-8">
<title>마우스가 움직인 곳의 위치 알아내기</title>
<style type="text/css">
div{ width: 200px; height: 150px;
    background-color: red; cursor: pointer;
    margin: 30px
}
</style>
<script type="text/javascript" src="../js/jquery.js"></script>
<script type="text/javascript">
    $(function(){
        $('div').mousemove(function(event){
            $('p:first').html(' 페이지를 기준으로 한 x 좌표 : ' + event.clientX
                + '<br/> 페이지를 기준으로 한 y 좌표 : ' + event.clientY + '<br/>');
            $('p:last').html(' 화면을 기준으로 한 x 좌표 : ' + event.screenX
                + '<br/> 화면을 기준으로 한 y 좌표 : ' + event.screenY + '<br/>');
        });
    });
</script>
</head>
<body>
    <div></div>
    <p></p> <p></p>
</body>
</html>
```

키보드 이벤트

❖ 키보드 이벤트

- keypress 이벤트는 한글로 사용할 수 없음

키보드 이벤트

이벤트 이름	설명
keydown	키보드가 눌려질 때 발생합니다..
keypress	글자가 입력될 때 발생합니다.
keyup	키보드가 떼어질 때 발생합니다.


- 예시

미투데이의 글쓰기

지금 내 생각을

마이마투데

150



✓ 올리기

키보드 이벤트

❖ 동적으로 글자 수 세기

- 키보드 이벤트와 키보드 이벤트의 주의할 점 숙지
- HTML 페이지 구성

```
<!DOCTYPE html>
<html>
<head>
  <script src="http://code.jquery.com/jquery-1.6.2.js"></script>
  <script>

    </script>
</head>
<body>
  <div>
    <p>지금 내 생각을</p>
    <h1>150</h1>
    <textarea cols="70" rows="5"></textarea>
  </div>
</body>
</html>
```


키보드 이벤트

❖ textarea 태그에 keyup 이벤트 연결

- keyup 이벤트 발생하면 글자 개수 받아 출력

```
<!DOCTYPE html>
<html>
<head>
  <script src="http://code.jquery.com/jquery-1.6.2.js"></script>
  <script>
    $(document).ready(function (event) {
      $('textarea').keyup(function () {
        // 남은 글자 수를 구합니다.
        var inputLength = $(this).val().length;
        var remain = 150 - inputLength;

        // 문서 객체에 입력합니다.
        $('h1').html(remain);
      });
    });
  </script>
</head>
```

키보드 이벤트

❖ 키보드 이벤트의 실행 순서

- 1. 사용자가 키보드 누름
- 2. keydown 이벤트 발생
- 3. 글자 입력
- 4. keypress 이벤트 발생
- 5. 사용자가 키보드에서 손을 땀
- 6. keyup 이벤트 발생
- keydown 이벤트가 발생한 순간에는 글자가 입력돼 있지 않음
- 입력한 글자 수를 표시해야 하므로 keyup 이벤트 사용

키보드 이벤트

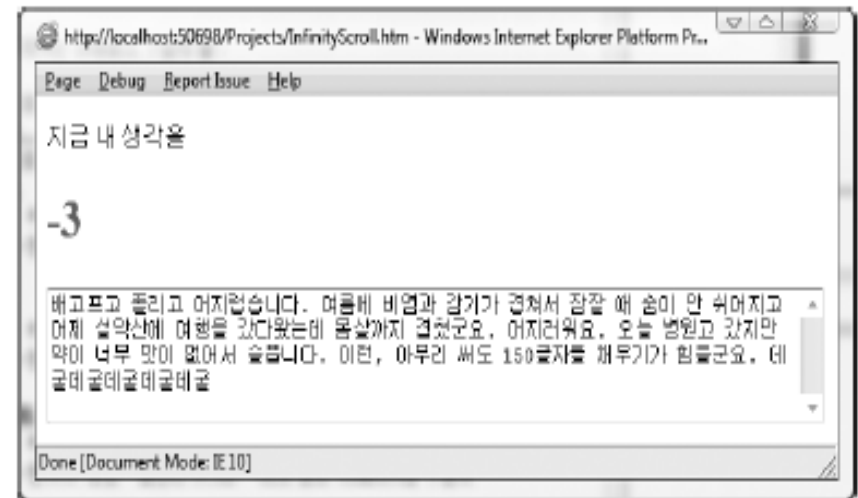
❖ 글자 개수에 따른 스타일 변경

```
<script>
$(document).ready(function (event) {
    $('textarea').keyup(function () {
        // 남은 글자 수를 구합니다.
        var inputLength = $(this).val().length;
        var remain = 150 - inputLength;

        // 문서 객체에 입력합니다.
        $('h1').html(remain);

        // 문서 객체의 색상을 변경합니다.
        if (remain >= 0) {
            $('h1').css('color', 'Black');
        } else {
            $('h1').css('color', 'Red');
        }
    });
});
</script>
```

실행 결과



윈도우 이벤트

❖ 윈도우 이벤트란?

- window 객체와 document 객체 이외에 img 태그 등이 사용할 수 있는 이벤트

윈도우 이벤트

이벤트	설명
ready	문서 객체가 준비를 완료합니다.
load	윈도우(문서 객체)를 불러들일 때 발생합니다.
unload	윈도우(문서 객체)를 닫을 때 발생합니다.
resize	윈도우의 크기를 변화시킬 때 발생합니다.
scroll	윈도우를 스크롤할 때 발생합니다.
error	에러가 있을 때 발생합니다.

윈도우 이벤트

❖ 윈도우 이벤트 사용한 무한 스크롤 구현

- 무한 스크롤은 최근 SNS와 같은 서비스에서 자주 볼 수 있는 효과
- Mixsh 사이트 참고 (<http://mixsh.com>)

<http://mixsh.com/>의 무한 스크롤



윈도우 이벤트

❖ jQuery 이벤트 사용해 간단하게 무한 스크롤 구현

- 1) 기본적인 HTML 페이지 구성

```
<!DOCTYPE html>
<html>
<head>
  <script src="http://code.jquery.com/jquery-1.7.js"></script>
  <script>
    $(document).ready(function () {

      });
  </script>
</head>
<body>

</body>
</html>
```

윈도우 이벤트

❖ jQuery 이벤트 사용해 간단하게 무한 스크롤 구현

- 2) window 객체에 scroll 이벤트 연결
 - window 객체는 별도의 선택자 사용하지 않음

```
<script src="http://code.jquery.com/jquery-1.7.js"></script>
<script>
    $(document).ready(function () {
        $(window).scroll(function () {

            });
    });
</script>
```

.scrollTop()은 선택한 요소의 스크롤바 수직 위치를 반환하거나 스크롤바 수직 위치

윈도우 이벤트

❖ jQuery 이벤트 사용해 간단하게 무한 스크롤 구현

■ window 객체에 scroll 이벤트 연결 원리

- 사용자가 마우스 스크롤 움직이면 무조건 scroll 이벤트 발생
- 무한 스크롤을 만들려면 화면 끝까지 스크롤이 도달했다는 사실 인식 필요
 - document 객체의 height 속성은 문서 전체의 높이 의미
 - 스크롤이 끝까지 내려가면

» window 객체 scrollTop + height = document 객체 높이

무한 스크롤 이벤트의 구성



윈도우 이벤트

❖ jQuery 이벤트 사용해 간단하게 무한 스크롤 구현

- 스크롤이 끝까지 닿으면 자동으로 h1 태그 10개 추가

```
<script src="http://code.jquery.com/jquery-1.7.js"></script>
<script>
    $(document).ready(function () {
        $(window).scroll(function () {
            var scrollHeight = $(window).scrollTop() + $(window).height();
            var documentHeight = $(document).height();
            if (scrollHeight == documentHeight) {
                for (var i = 0; i < 10; i++) {
                    $('<h1>Infinity Scroll</h1>').appendTo('body');
                }
            }
        });
    });
</script>
```

윈도우 이벤트

❖ jQuery 이벤트 사용해 간단하게 무한 스크롤 구현

■ 전체 스크립트 코드

```
<script>      // 무한 스크롤 부분
$(document).ready(function () {      // 스크롤 이벤트 발생 시
    $(window).scroll(function () {      // 필요한 변수를 구합니다.
        var scrollHeight = $(window).scrollTop() + $(window).height();
        var documentHeight = $(document).height();
        // 스크롤의 높이와 문서의 높이가 같을 때
        if (scrollHeight == documentHeight) {
            for (var i = 0; i < 10; i++) {
                $('<h1>Infinity Scroll</h1>').appendTo('body');
            }
        }
    });
});
// 테스트를 위해 내부에 공간을 채워둡니다.
$(document).ready(function () {
    for (var i = 0; i < 20; i++) {
        $('<h1>Infinity Scroll</h1>').appendTo('body');
    }
});
</script>
```

윈도우 이벤트

❖ 스크롤 구현 결과

- 두 번째 `$(document).ready()` 이벤트에서 `h1` 태그 20개 생성
 - 현재 문서에 아무 내용도 없으니 스크롤이 생기지 않기 때문
- 무한 스크롤을 바로 모바일에 적용하기 어려움
 - 모바일 디바이스에 따라 문서 객체가 60픽셀에서 200픽셀 정도 크게 나옴
 - 직접 장치를 확인하고 처리해야 함

끝없이 계속되는 스크롤



입력 양식 이벤트

❖ 입력 양식 이벤트

입력 양식 이벤트

이벤트 이름	설명
change	입력 양식의 내용을 변경할 때 발생합니다.
focus	입력 양식에 초점을 맞추면 발생합니다.
focusin	입력 양식에 초점이 맞춰지기 바로 전에 발생합니다.
focusout	입력 양식에 초점이 사라지기 바로 전에 발생합니다.
blur	입력 양식에 초점이 사라지면 발생합니다.
select	입력 양식을 선택할 때 발생합니다(input[type=text]태그와 textarea 태그 제외).
submit	submit 버튼을 누르면 발생합니다.
reset	reset 버튼을 누르면 발생합니다.

입력 양식 이벤트

❖ submit 이벤트

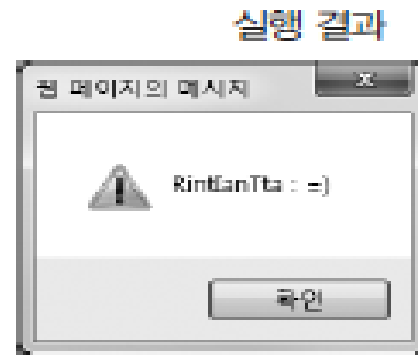
- 입력 양식이 있어야 하므로 body 태그 구성

```
<body>
  <form id="my_form">
    <table>
      <tr>
        <td>이름: </td>
        <td><input type="text" name="name" id="name"/></td>
      </tr>
      <tr>
        <td>비밀번호: </td>
        <td><input type="password" name="password" id="password"/></td>
      </tr>
    </table>
    <input type="submit" value="제출"/>
  </form>
</body>
```

입력 양식 이벤트

❖ submit 이벤트

- form 태그에서 발생하는 이벤트
 - form 객체에 submit() 메서드 연결
 - 유효성 검사를 할 때는 기본 이벤트 제거해야 함



```
<script>
$(document).ready(function () {
    $('#my_form').submit(function (event) {
        // 입력 양식의 value를 가져옵니다.
        var name = $('#name').val();
        var password = $('#password').val();
        // 출력합니다.
        alert(name + ' : ' + password);

        // 기본 이벤트를 제거합니다.
        event.preventDefault();
    });
});
</script>
```

입력 양식 이벤트

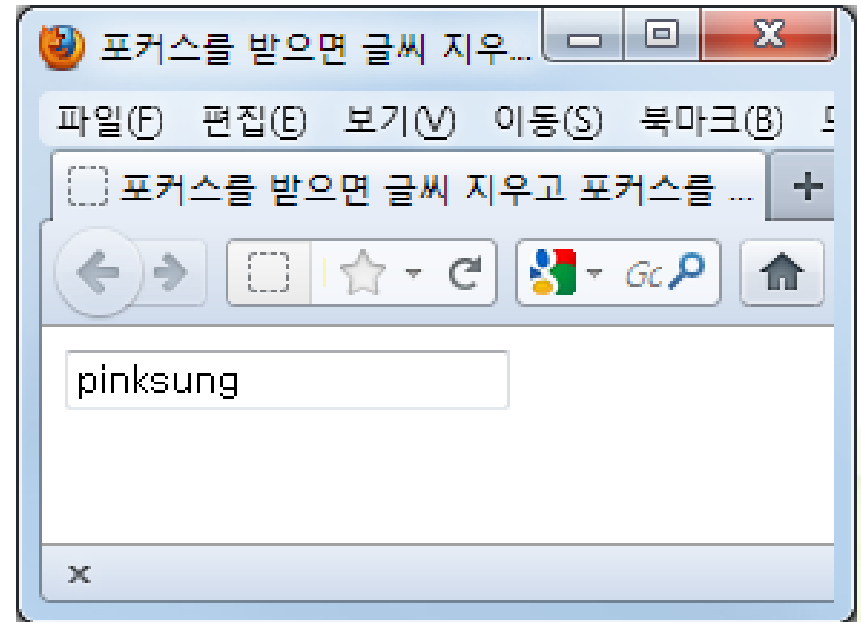
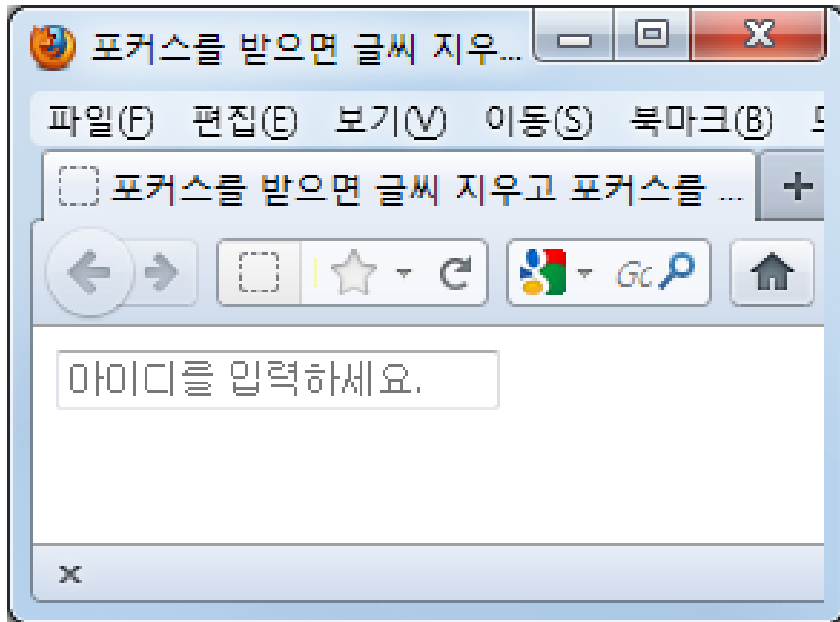
❖ change 이벤트

- type 속성이 checkbox와 radio인 input 태그의
- 상태 변경하는 이벤트
- click 이벤트가 아님

```
<!DOCTYPE html><html><head>
<script src="http://code.jquery.com/jquery-1.7.js"></script>
<script>
    $(document).ready(function () {
        $('#all_check').change(function(event) {
            if (this.checked) {
                $('div').children().prop('checked',this.checked);
                // $(' #check_item').children().attr('checked', 'checked');
            } else {
                // // $(' #check_item').children().removeAttr('checked');
                $(' #check_item').children().prop('checked', false);
            }
        });
    });
</script></head><body>
<input type="checkbox" id="all_check"/>
<label>All</label>
```

```
<div id="check_item">
    <input type="checkbox"/>
    <label>A Option</label>
    <input type="checkbox"/>
    <label>B Option</label>
    <input type="checkbox"/>
    <label>C Option</label>
</div>
</body>
</html>
```

포커스를 받으면 글씨지우고 포커스를 잃으면 글씨나타내기



- **focus()**

- 포커스를 받을 때 발생하는 focus 이벤트를 붙인다.

- **blur()**

- 포커스를 잃을 때 발생하는 blur 이벤트를 붙인다.

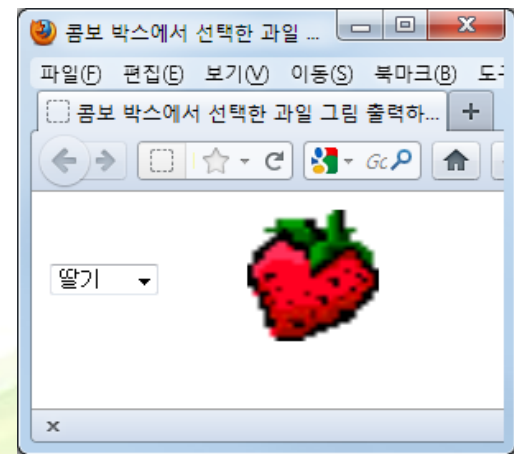
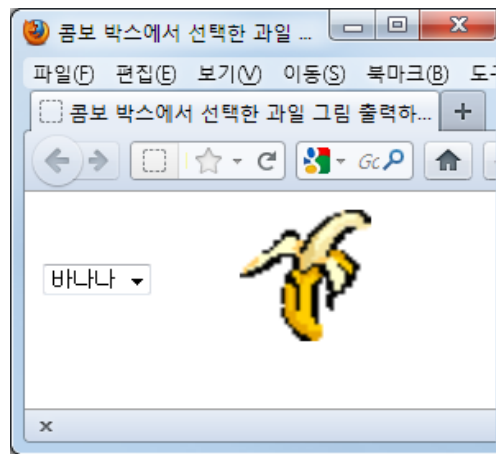
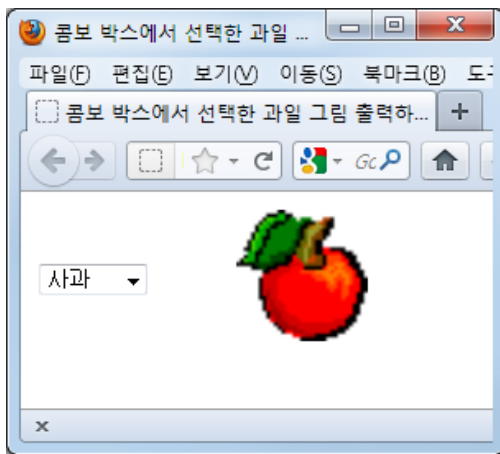
포커스를 받으면 글씨지우고 포커스를 잃으면 글씨나타내기

```
$('#input').val("아이디를 입력하세요. ").css("color", "#777");  
$('#input').one('focus', function(event) {  
    $(event.target).val("").css("color", "#000");  
});  
$('#input').blur(function(event) {  
    if($(event.target).val()==""){  
        $(event.target).val("아이디를 입력하세요. ").css("color",  
            "#777");  
    }  
});
```

```
<!DOCTYPE html><html><head>
<meta charset="UTF-8">
<title>포커스를 받으면 글씨 지우고 포커스를 잃으면 글씨 나타내기</title>
<script type="text/javascript" src="../js/jquery.js"></script>
<script type="text/javascript">
    $(function() {
        $('input').val('ID를 입력하세요').css('color','red');
        $('input').bind('focus',function(){
            $(this).val('').css('color','blue');
        });
        $('input').blur(function(){
            if ($(this).val()=='') {
                $(this).val('ID를 입력하세요').css('color','red');
            }
        });
    });
</script>
</head>
<body>
    <INPUT TYPE="text" id="ID">
</body>
</html>
```

콤보 박스에서 선택한 과일 그림 출력하기

```
$('#sel').change(function(event) {  
    var imgSrc = $(event.target).val();  
    $('#img').attr('src', imgSrc);  
}).change();
```



■ change(function)

- 값이 변경될 때 발생하는 change 이벤트를 붙인다.

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>콤보 박스에서 선택한 파일 그림 출력하기</title>
<style type="text/css">
img{
    width: 80px; height: 80px; margin-left: 50px;
}
</style>
<script type="text/javascript" src="../../js/jquery.js"></script>
<script type="text/javascript">
    $(function() {
        $('#sel').change(function(event) {
            var imgSrc = $(event.target).val();
            $('#img').attr('src', imgSrc);
        }).change();
    });
</script>
</head>
```

```
<body>
  <form name="frm">
    <table>
      <tr>
        <td>
          <select id="sel">
            <option value="../images/strawberry.gif">딸기
            <option value="../images/banana.gif">바나나
            <option value="../images/apple.gif" selected>사과
          </select>
        </td>
        <td><img id="img"></td>
      </tr>
    </table>
  </form>
</body>
</html>
```