

어트리뷰트

강사 : 강병준

JQUERY 속성 선택자

❖ 속성 선택자

- 기본 선택자 뒤에 붙여 사용
 - 요소[속성] : 특정 속성을 가지는 있는 문서 선택
 - 요소[속성=값] : 속성의 값이 일치
 - 요소[속성*=값] : 속성의 값에 포함
 - 요소[속성^=값] : 속성 값 시작
 - 요소[속성\$=값] : 속성 값 끝
-
- \$('input[type=text]') : <input type= “text” />

어트리뷰트를 조건으로 제시하여 엘리먼트를 찾기

$E[A]$, $E[A=V]$

```
$('span[id]')
```

어트리뷰트 id를 가지는 모든 span 엘리먼트를 찾는다.

```
$('span[id="simpletext1"]')
```

어트리뷰트 id의 값이 simpletext1인 모든 span 엘리먼트를 찾는다.

```
$('span[id!="simpletext1"]')
```

어트리뷰트 id의 값이 simpletext1가 아닌 모든 span 엘리먼트를 찾는다.

어트리뷰트를 조건으로 제시하여 엘리먼트를 찾기

$E[A^=V]$, $E[A\$=V]$, $E[A*=V]$

```
$( 'span[id^="complex"]' )
```

어트리뷰트 id의 값이 complex로 시작하는 모든 span 엘리먼트를 찾는다.

```
$( 'span[id$="text1"]' )
```

어트리뷰트 id의 값이 text1으로 끝나는 모든 span 엘리먼트를 찾는다.

```
$( 'span[id*="text"]' )
```

값에 text를 포함하는 어트리뷰트 id를 가지고 있는 모든 span 엘리먼트를 찾는다.

JQUERY 속성 선택자

❖ 속성 선택자 사용 예제 코드

attText.html

```
<!DOCTYPE html>
<html>
<head>
  <script src="http://code.jquery.com/jquery-1.7.js"></script>
  <script>
    $(document).ready(function () {
      $('input[type=text]').val('Hello jQuery..!');
    });
  </script>
</head>
<body>
  <input type="text" /><br>
  <input type="password" /><br>
  <input type="radio" /><br>
  <input type="checkbox" /><br>
  <input type="file" />
</body>
</html>
```

Hello jQuery.. !



찾아보기...

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>어트리뷰트를 조건으로 제시하여 엘리먼트를 찾기-E[A], E[A=V], E[A^=V], E[A$=V],
E[A*=V]</title>
<style type="text/css">
.spotlight { background-color: #ff0}
.redtext { color: #f00}
.bluetext { color: #00f}
.largetext { font-size: 30pt}
.italictext { font-style: italic}
.box { border: 3pt solid #f00}
</style>
<script src="../../js/jquery.js" type="text/javascript"></script>
<script type="text/javascript">
    var output = function(selector) {
        var resultText = "";
        console.log("Wn" + selector + "로 검색된 엘리먼트의 갯수 " + $(selector).length);
        $(selector).each(function() {
            resultText+= $(this).text() + "Wt";
        });

        console.log(resultText + "Wn");
    };
};
```

```
$(document).ready(function() {
    $('span[id]').addClass('spotlight');
    $('span[id="simpletext1"]').addClass('redtext');
    $('span[id!="simpletext1"]').addClass('bluetext');
    $('span[id^="complex"]').addClass('italictext');
    $('span[id$="text1"]').addClass('largetext');
    $('span[id*="text"]').addClass('box');

    output("span[id]");
    output("span[id='simpletext1']");
    output("span[id!='simpletext1']");
    output("span[id^='complex']");
    output("span[id$='text1']");
    output("span[id*='text']");
});
</script>
</head>
<body>
    <span id="simpletext1">simple</span>
    <span class="simpletext2">jQuery</span>
    <span id="complextext1">basic</span>
    <span id="complextext2">example</span>
    <span id="complex">book</span>
</body>
</html>
```

JQUERY 입력 양식 필터 선택자

❖ 속성 선택자 사용

- input 태그의 type 속성 따라 문서 객체 선택 가능, 필터 선택자 사용
 - 필터 선택자는 기본 선택자 뒤에 사용
 - 'Input:button' 의 형태로 선택

선택자 형태

설명

요소:button input태그 중 type속성이 button인 문서객체인 button태그 선택

요소:checkbox input태그 중 type속성이 check인 문서 객체를 선택

요소:file input태그 중 type속성이 file인 문서 객체를 선택

요소:image input태그 중 type속성이 image인 문서 객체를 선택

요소:password input태그 중 type속성이 password인 문서 객체를 선택

요소:radio input태그 중 type속성이 radio인 문서 객체를 선택

요소:reset input태그 중 type속성이 reset인 문서 객체를 선택

요소:submit input태그 중 type속성이 submit인 문서 객체를 선택

요소:text input태그 중 type속성이 text인 문서 객체를 선택

JQUERY 입력 양식 필터 선택자

❖ 속성 선택자 사용

선택자 형태	설명
--------	----

요소:checked	체크된 입력 양식 선택
------------	--------------

요소:disabled	비활성화된 입력 양식 선택
-------------	----------------

요소:enabled	활성화된 입력 양식 선택
------------	---------------

요소:focus	초점이 맞춰져 있는 입력 양식 선택
----------	---------------------

요소:input	모든 입력 양식 선택
----------	-------------

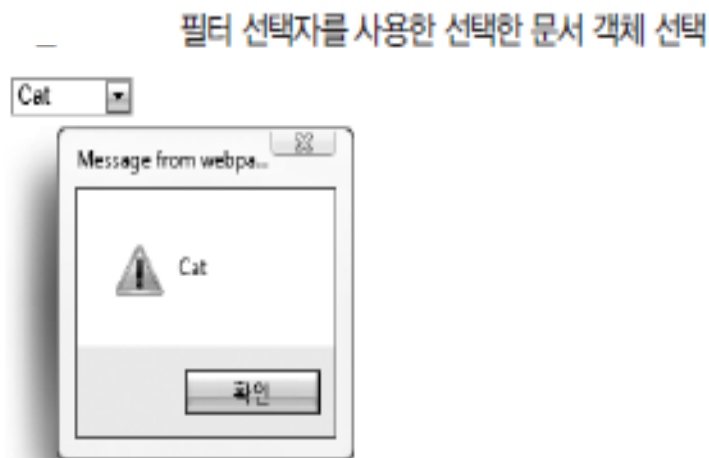
요소:selected option	객체중에서 선택된 태그 선택
--------------------	-----------------

JQUERY 입력 양식 필터 선택자

❖ 속성 선택자 사용 예제

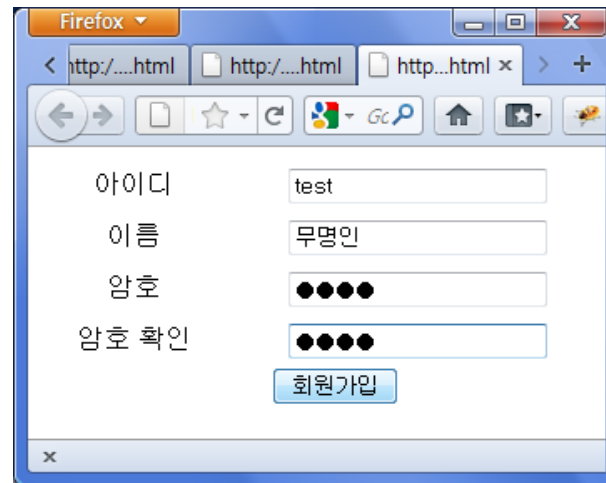
- setTimeout() 함수 사용해 웹 페이지 실행 → 5초 정도 후 선택자 사용
- body 태그 구성 setTime.html

```
<head>
<script src="http://code.jquery.com/jquery-1.7.js"></script>
<script>
    $(document).ready(function () {
        setTimeout(function () {
            var value = $('select > option:selected').val();
            alert(value);
        }, 5000);
        /* setInterval( function() {
        var va = $('select option:selected').val();
        alert(va);
        }, 5000); */
    });
</script></head>
<body>
<select>
    <option>Apple</option>    <option>Bag</option>
    <option>Cat</option>     <option>Dog</option>
    <option>Elephant</option>
</select>
</body>
```



JQUERY 정의 필터 셀렉터를 이용하여 폼 요소 다루기

:button, :text, :password



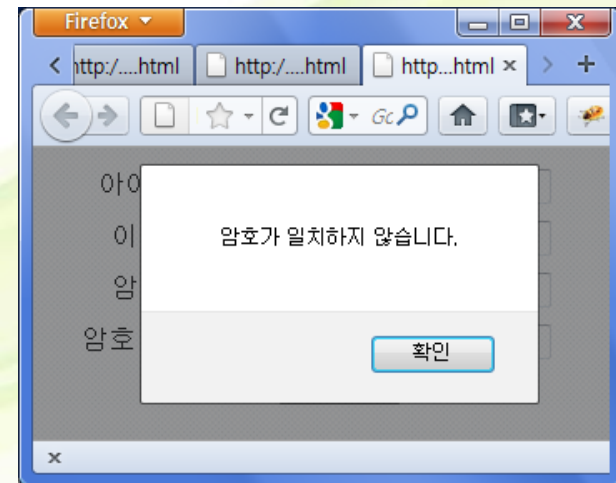
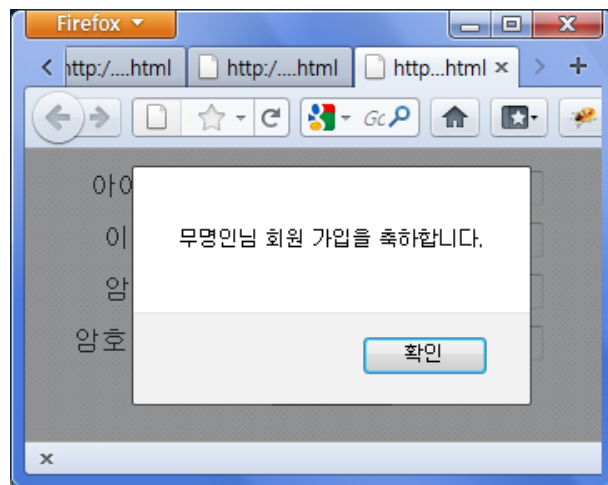
아이디 test

이름 무명인

암호 ●●●●

암호 확인 ●●●●

회원가입



JQUERY 정의 필터 셀렉터를 이용하여 폼 요소 다루기

:button, :text, :password

```
$('#regBtn:button')
```

엘리먼트 중 type 어트리뷰트 값이 button인 엘리먼트 중에서도 id 어트리뷰트 값이 regBtn인 엘리먼트를 찾는다.

```
$('#userId:text')
```

type 어트리뷰트 값이 text인 엘리먼트 중에서도 id 어트리뷰트 값이 userId인 엘리먼트를 찾는다.

JQUERY 정의 필터 셀렉터를 이용하여 폼 요소 다루기

- `click`
 - 버튼이 눌리면 인자에 기술한 함수가 자동 호출하여 이벤트를 처리하도록 하는 jQuery가 제공하는 함수
- `attr(name)`
 - 어트리뷰트 값을 얻어오기
- `attr(name, value)`
 - 새로운 어트리뷰트 값을 설정하기
- `val()`
 - 값을 얻어오기

JQUERY 정의 필터 셀렉터를 이용하여 폼 요소 다루기

jQuery 정의 필터 셀렉터

셀렉터	설명
:animated	현재 애니메이션이 적용되고 있는 엘리먼트를 선택한다.
:button	모든 버튼을 선택한다(input[type=submit], input[type=reset], input[type=button], button).
:checkbox	체크박스 엘리먼트만 선택한다.
:checked	선택된 체크박스나 라디오 버튼만을 선택한다(CSS에서 지원).
:contains(foo)	텍스트 foo를 포함하는 엘리먼트만 선택한다.
:disabled	인터페이스에서 비활성화 상태인 모든 폼 엘리먼트를 선택한다(CSS에서 지원).
:enabled	인터페이스에서 활성화 상태인 모든 폼 엘리먼트를 선택한다(CSS에서 지원).
:file	모든 파일 엘리먼트를 선택한다(input[type=file]).
:header	헤더 엘리먼트만 선택한다. 예를 들어<h1>부터 <h6>까지의 엘리먼트를 선택한다.

JQUERY 정의 필터 셀렉터를 이용하여 폼 요소 다루기

jQuery 정의 필터 셀렉터

셀렉터	설명
:image	폼 이미지를 선택한다(input[type=image]).
:input	폼 엘리먼트만 선택한다(input, select, textarea, button).
:not(filter)	필터의 값을 반대로 변경한다.
:parent	빈 엘리먼트를 제외하고, 텍스트도 포함해서 자식 엘리먼트를 가지는 엘리먼트를 선택한다.
:password	패스워드 엘리먼트만 선택한다(input[type=password]).
:radio	라디오 버튼 엘리먼트만 선택한다(input[type=radio]).
:reset	리셋 버튼을 선택한다(input[type=reset] 이나 button[type=reset]).
:selected	선택된 엘리먼트만 선택한다.
:submit	전송 버튼을 선택한다(button[type=submit] 이나 input[type=submit]).
:text	텍스트 엘리먼트만 선택한다(input[type=text]).
:visible	보이는(visible) 엘리먼트만 선택한다.

```

<!DOCTYPE html><html><head><meta charset="UTF-8">
<title>jQuery 정의 필터 셀렉터를 이용하여 폼 요소 다루기-
:button, :text, :password</title>
<style type="text/css">
    div { padding: 5px 5px 5px 5px }
    .item{float:left; width:100px; }
</style>
<script src="../js/jquery.js" type="text/javascript"></script>
<script type="text/javascript">
    $(document).ready(function() {
        $('#regBtn:button').click(function(){
            invalidItem(); clearInput();
        });
        function invalidItem(){
            if ($("#userId:text").val()==""){ alert("아이디를 입력하세요."); return; }
            var userName = $('#userName:text').val();
            if(userName == ""){ alert("이름을 입력하세요."); return; }
            if( $("#pwd:password").val() =="" || ($("#pwdConfirm:password").val() == "")){
                alert("암호를 입력하세요."); return;
            }
            if($("#pwd:password").val() !=
                $("#pwdConfirm:password").val()) {
                alert("암호가 일치하지 않습니다."); return;
            }
            alert(userName+"님 회원 가입을 축하합니다.");
        }
    }

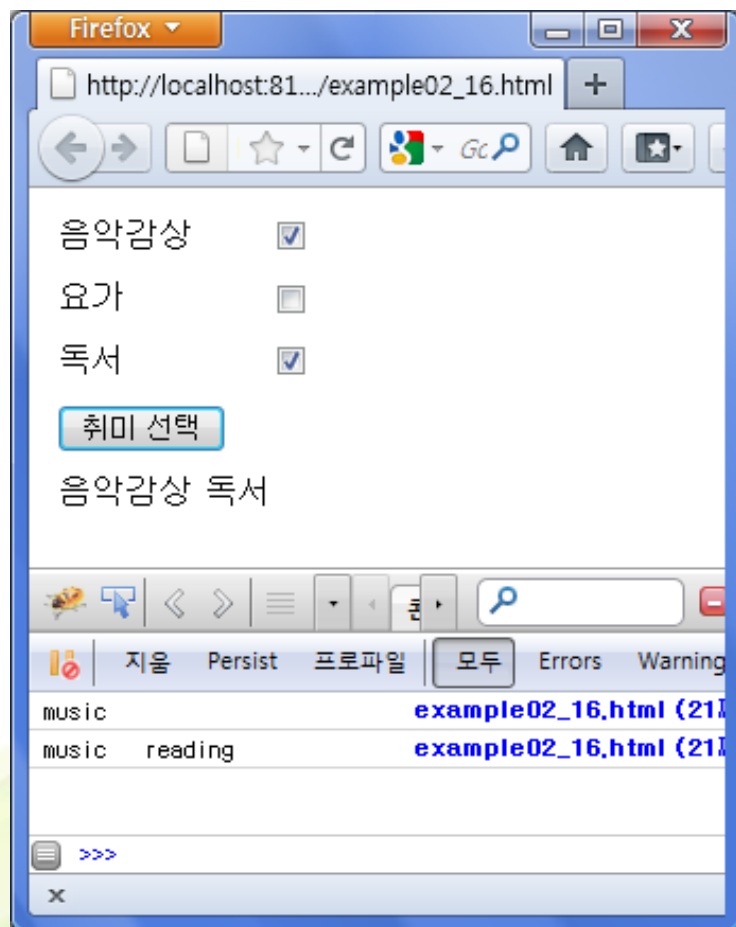
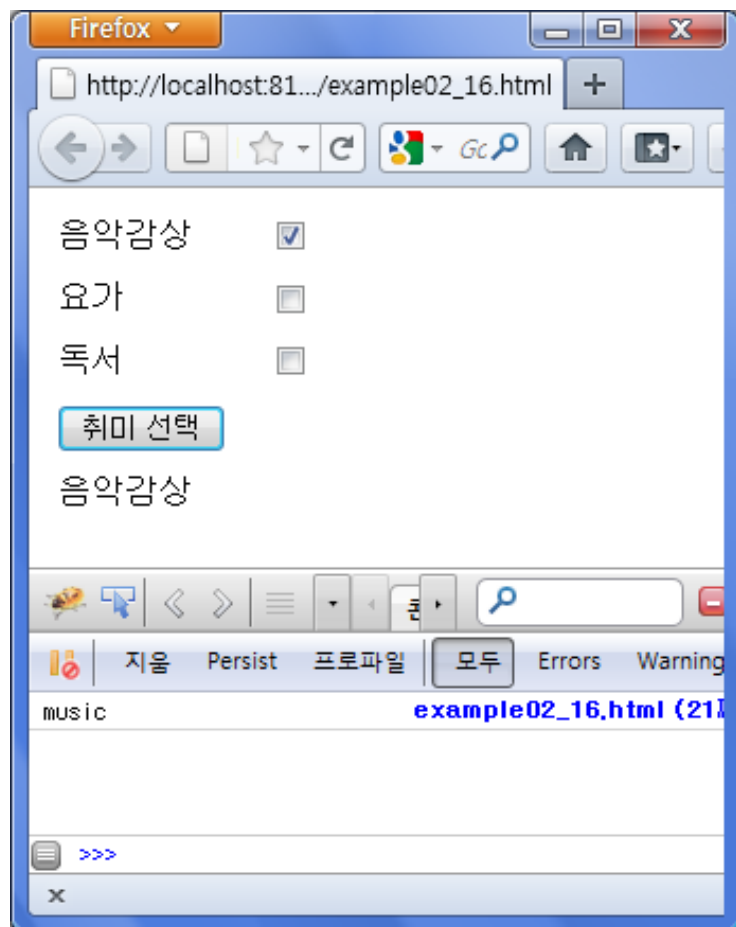
```



```
function clearInput(){
    $('#userId:text').val('');
    $('#userName:text').val('');
    $('#pwd:password').val('');
    $('#pwdConfirm:password').val('');
}
});
</script>
</head>
<body>
    <center>
        <div> <span class="item">아이디 * </span>
            <input id="userId" type="text"></div>
        <div> <span class="item">이름 * </span>
            <input id="userName" type="text"></div>
        <div> <span class="item">암호 * </span>
            <input id="pwd" type="password"></div>
        <div> <span class="item">암호 확인 * </span>
            <input id="pwdConfirm" type="password"></div>
        <input type="button" id="regBtn" value="회원가입">
    </center>
</body>
</html>
```

여러 개의 취미를 체크박스에서 선택하기

:checkbox



여러 개의 취미를 체크박스에서 선택하기

:checkbox

```
$("#input[name='hobby']:checkbox:checked")
```

name 어트리뷰트 값이 'hobby'인 체크박스 중에서 체크된 엘리먼트 집합을 얻어온다.

■ prev()

- 엘리먼트의 형제 엘리먼트 가운데 바로 이전 엘리먼트의 집합을 반환한다.

■ html(htmlString)

- 인자로 각 엘리먼트에 지정할 HTML 문자열을 지정하면 선택된 엘리먼트의 기존 내용을 새로운 내용으로 대체한다.

```
<!DOCTYPE html>
<html><head><meta charset="UTF-8">
<title>여러 개의 취미를 체크박스에서 선택하기-:checkbox</title>
<style>
div { padding: 5px 5px 5px 5px }
.item { float: left; width: 100px; }
</style>
<script src="../../js/jquery.js" type="text/javascript"></script>
<script type="text/javascript">
$(document).ready(function() {
    $('#selectBtn:button').click(function(){
        var checkedList = "";
        var checkedConsole = "";
        $("input[name='hobby']:checkbox:checked").each(function() {
            checkedList += $(this).prev().text() + "Wt";
            checkedConsole += $(this).val() + "Wt";
        });
        $('#contents').html(checkedList);
        console.log(checkedConsole);
    });
});
</script>
</head>
```

```
<body>
<div>
<span class="item"> 음악감상 </span>
<input type="checkbox" name="hobby" value="music">
</div>
<div>
<span class="item"> 요가 </span>
<input type="checkbox" name="hobby" value="yoga">
</div>
<div>
<span class="item"> 독서 </span>
<input type="checkbox" name="hobby" value="reading">
</div>
<div>
<input type="button" id="selectBtn" value="취미 선택"></div>
<div id="contents"> 결과 </div>
</body>
</html>
```

어트리뷰트란 무엇인가?

```
<a href= “#” >  
    
</a>
```

- img
 - 태그명
- src, title, alt
 - 어트리뷰트
- 어트리뷰트는 이름과 값으로 이루어져 있음
- src 어트리뷰트값을 얻어오거나 src 어트리뷰트값을 대
경로명으로 변경하여 설정하기 위한 방법

엘리먼트의 어트리뷰트 값을 가져오고, 설정하기

`attr(attributeName), attr(attributeName, value)`

```
$("#img").attr("alt")
```

```
$("#img").attr("src", "../images/im0.jpg")  
          .attr("title", "돌이군")  
          .attr("alt", "puppy picture");
```

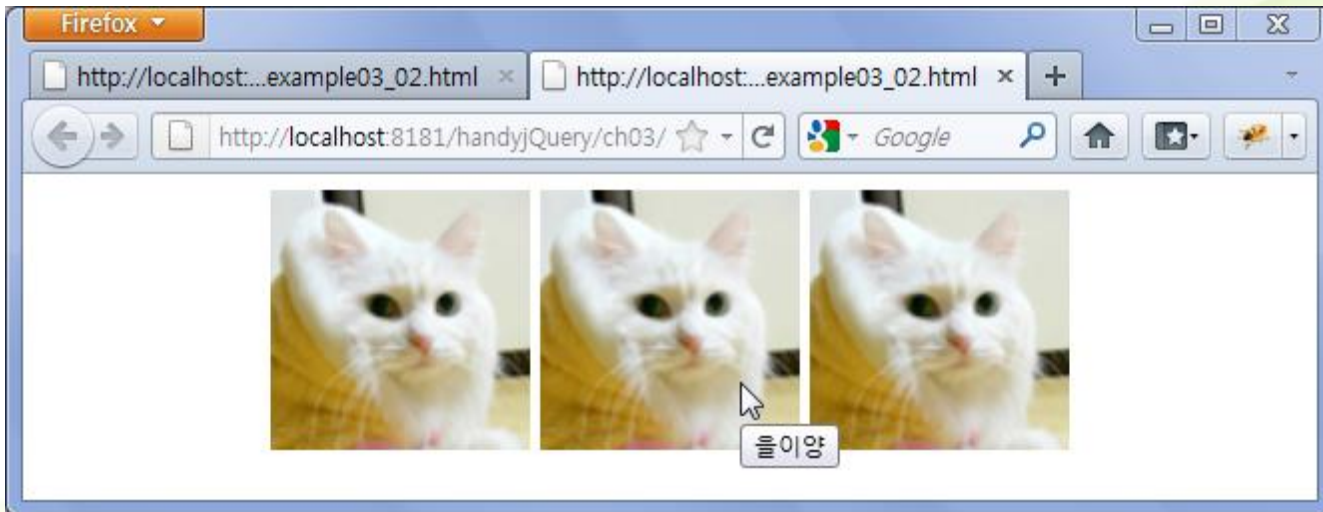


```
<!DOCTYPE html><html><head>
<meta charset="UTF-8">
<title>엘리먼트의 어트리뷰트 값을 가져오고, 설정하기</title>
<style>
    img { padding:20px; }
    div { color: blue; font-size: 30px }
</style>
<script src="../../js/jquery.js" type="text/javascript"></script>
<script type="text/javascript">
    $(document).ready(function() {
        $("img").attr("src", "../../images/im0.jpg")
            .attr("title", "똥이군")
            .attr("alt", "puppy picture");
        $("div").text($("img").attr("alt"));
    });
</script></head>
<body>
<center>
    <img /> <img />
    <div><B> 어트리뷰트 연습</B></div>
</center>
</body>
</html>
```


엘리먼트 집합들을 한 번에 처리하기

attr({ attributeName : value, attributeName : value, })

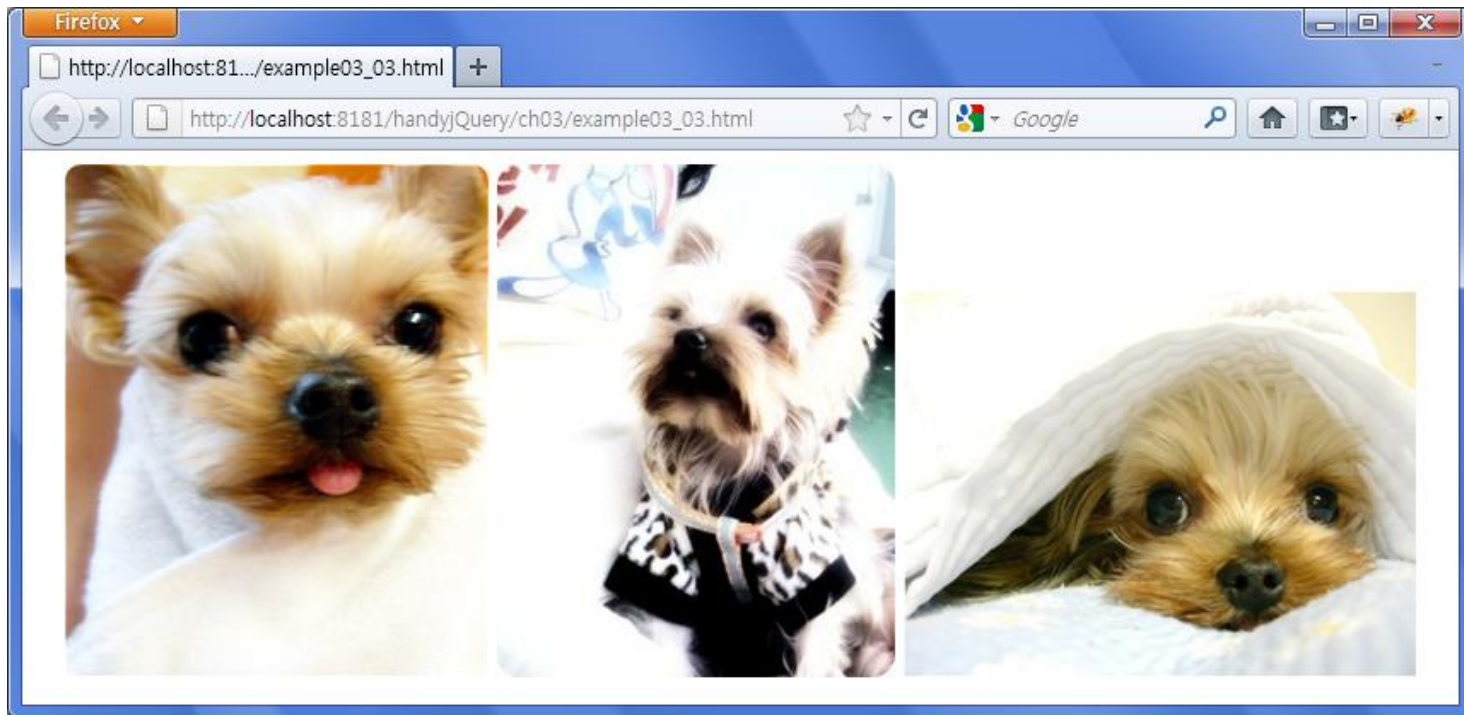
```
$("#img").attr( {  
    src: "../images/im4.jpg",  
    title: "을이양",  
    alt: "cat picture",  
    width: "130px",  
    height: "130px"  
});
```



```
<!DOCTYPE html><html><head><meta charset="UTF-8">
<title>엘리먼트 집합들을 한 번에 처리하기</title>
<script src="../js/jquery.js" type="text/javascript"></script>
<script type="text/javascript">
    $(function() {
/*          $('img').attr({
                        src: '../images/im4.jpg',
                        title: '용팔이', alt: '누구세요',
                        width: '200px', height: '200px'
                    }); */
        $('img').each(function(index) {
            $(this).attr({
                src: '../images/im'+index+'.jpg',
                title: '용팔이', alt: '누구세요',
                width: '200px', height: '200px'
            });
        });
    });</script>
</head><body>
    <center>
        <img /> <img /> <img />
    </center>
</body></html>
```

엘리먼트의 어트리뷰트 값을 콜백 함수에서 얻어 와서 설정

```
$("#img").attr("src", function() {  
    return "../images/" + this.title;  
});
```



엘리먼트의 어트리뷰트 값을 콜백 함수에서 얻어 와서 설정

`attr(attributeName, function())`

- value 인자 위치에 함수를 기술하면
- 셀렉터에 의해 검색된 엘리먼트 집합에 속한 개별의 엘리먼트마다 전달된 함수가 호출되며 함수의 반환값이 어트리뷰트의 값으로 사용
- 개별 엘리먼트가 함수를 호출할 때 해당 엘리먼트가 `this`에 할당됨

엘리먼트의 어트리뷰트 값을 콜백 함수에서 얻어 와서 설정

`attr(attributeName, function())`

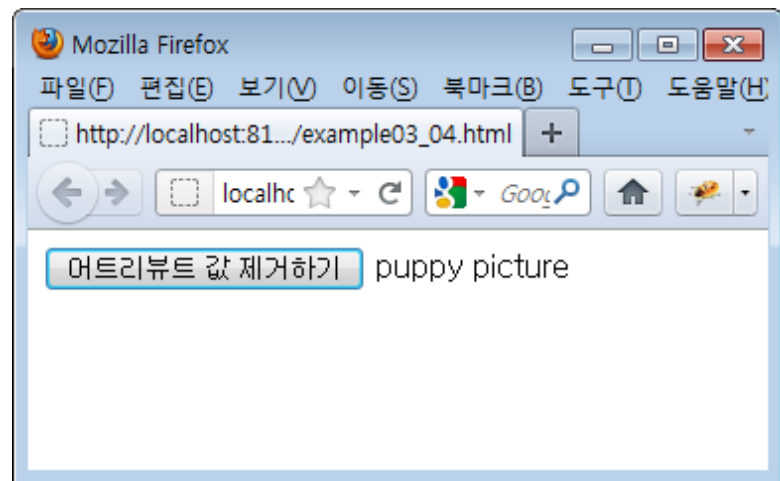
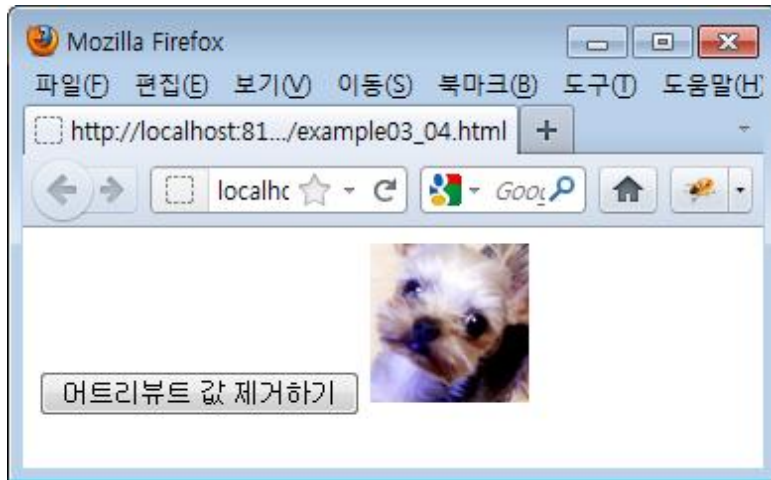
```
$("#img").attr("title", function(index) {  
    return index + "번 엘리먼트로 타이틀은 " + this.title + "이다.";  
});
```

- 함수에 `index`를 기술하면 0부터 시작되는 인덱스가 함수로 전달됨

```
<!DOCTYPE html><html><head>
<meta charset="UTF-8">
<title>엘리먼트의 어트리뷰트 값을 콜백 함수에서 얻어 와서 설정하기</title>
<script src="../js/jquery.js" type="text/javascript"></script>
<script type="text/javascript">
    $(document).ready(function() {
        $("img").attr({width: "100px",
            height: "100px"});
        $("img").attr("src", function() {
            return "../images/" + this.title;
        });
        $("img").attr("title", function(index) {
            return index + "번 엘리먼트로 타이틀은 " + this.title + "이
다.";
        });
    });
</script></head>
<body>
<center>
    <img title="image1.jpg"/>
    <img title="image2.jpg"/>
    <img title="image3.jpg"/>
</center>
</body></html>
```

엘리먼트의 어트리뷰트 값 제거하기

```
.removeAttr("src");
```



엘리먼트의 어트리뷰트 값 제거하기

`removeAttr(attributeName)`

- **alt 어트리뷰트의 역할**

- 이미지를 찾지 못했을 경우 이를 대체해서 출력되는 텍스트

- **click(fn)**

- 대상 엘리먼트에 마우스 포인터를 위치시키고, 마우스 버튼을 누르고 떼었을 때 인자로 기술한 함수(이벤트 동작 함수)를 호출한다.
- fn : 클릭 이벤트가 발생했을 때 실행할 함수

- **next()**

- 해당 엘리먼트 뒤에 위치하는 인접한 형제 엘리먼트 한 개만 찾기


```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>엘리먼트의 어트리뷰트 값 제거하기</title>
<script src="../../js/jquery.js" type="text/javascript"></script>
<script type="text/javascript">
    $(function(){
        $("button").click(function () {
            $(this).next().removeAttr("src");
        });
    });
</script>
</head>
<body>
    <button> 어트리뷰트 값 제거하기 </button>
    
</body>
</html>
```

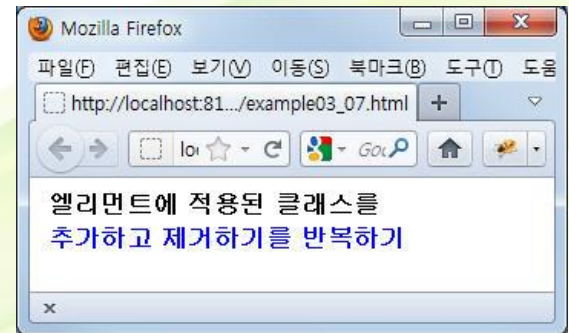
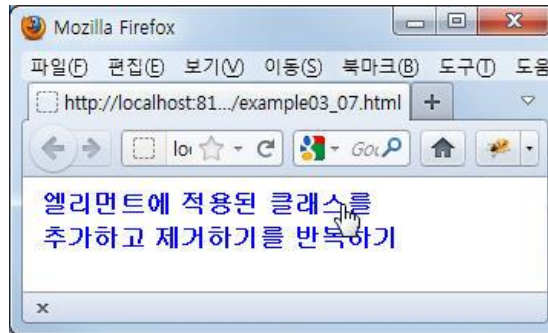
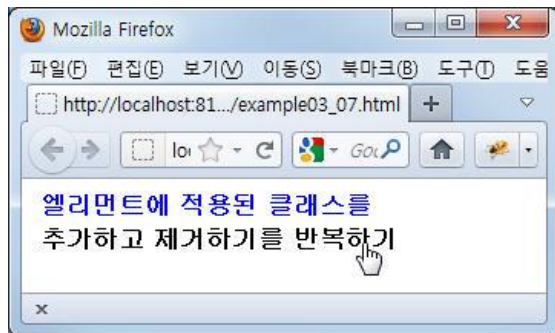
```
<!DOCTYPE html><html><head><meta charset="UTF-8"> <title>Insert title here</title>
<style> p { margin: 4px; font-size: 20px; font-weight: bolder;}
        .red { color: red;}
        .under { text-decoration: underline;}
        .highlight { background: yellow;}
</style>
<script type="text/javascript" src="../../js/jquery2.js"></script>
<script type="text/javascript">
    $(function() {
        $("p").addClass("red under");
        $("#removeClass").click(function() {
            $("p:even").removeClass("red under");
        });
        $("#changeClass").click(function() {
            $("p:odd").removeClass("red under").addClass('highlight');
        });
    });
</script></head><body>
    <button id="removeClass">removeClass</button>
    <button id="changeClass">changeClass</button>
    <p>엘리먼트로부터</p> <p>클래스를</p> <p>추가하고</p>
    <p>제거하는</p> <p>연습하기</p>
</body></html>
```

엘리먼트에 해당 클래스가 있으면 제거하고, 없으면 추가

```
toggleClass(className)
```

```
$(this).toggleClass("blue");
```

현재의 클래스 값에 따라서 클래스를 추가하거나 지운다.



```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<meta charset="UTF-8">
```

```
<title>엘리먼트에 해당 클래스가 있으면 제거하고, 없으면 추가하기</title>
```

```
<style>
```

```
p {    margin: 4px;    font-size: 16px;
      font-weight: bolder;    cursor: pointer; }
```

```
.blue {    color: blue; }
```

```
</style>
```

```
<script src="../../js/jquery.js" type="text/javascript"></script>
```

```
<script type="text/javascript">
```

```
$(function() {
    $("p").click(function () {
        $(this).toggleClass("blue");
    });
});
```

```
</script>
```

```
</head>
```

```
<body>
```

```
<p class="blue">엘리먼트에 적용된 클래스를</p>
```

```
<p>추가하고 제거하기를 반복하기</p>
```

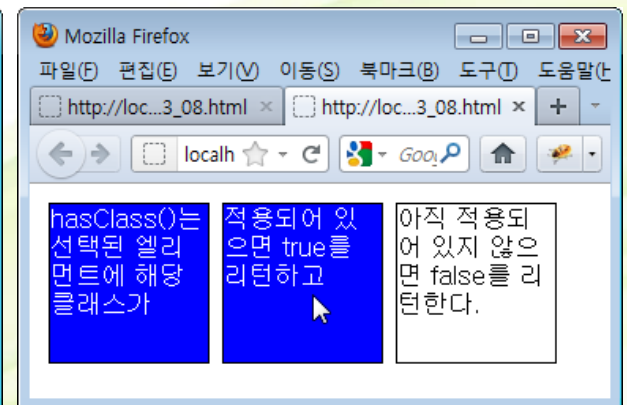
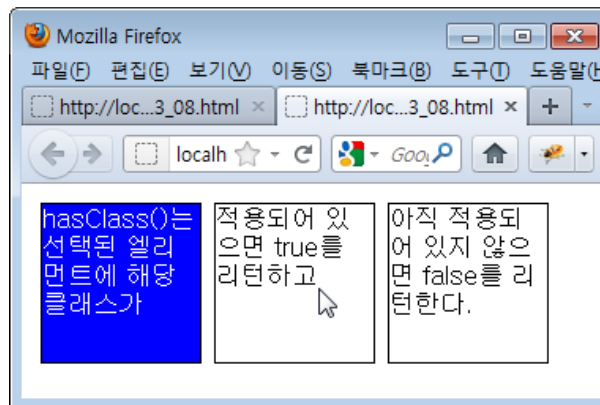
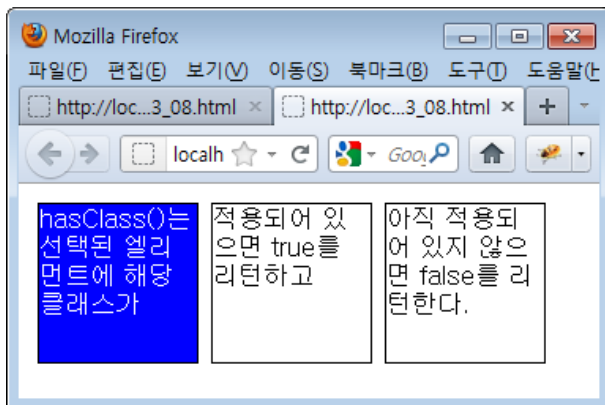
```
</body></html>
```

엘리먼트에 클래스 존재 여부 확인하기

hasClass(className)

```
$("#p:first").hasClass("red")
```

클래스가 지정되어 있는지를 확인한다.

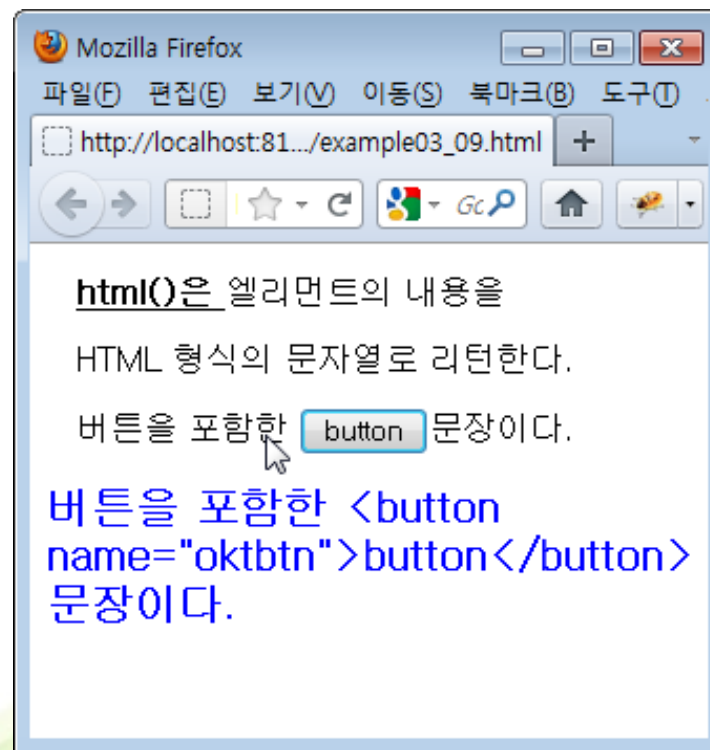


```
<!DOCTYPE html><html><head><meta charset="UTF-8">
<title>엘리먼트에 클래스 존재 여부 확인하기</title>
<style>
  div{ width:100px; height:100px; border:1px solid #000000;
    cursor: pointer; margin: 4px; float:left}
  .toggleStyle { background-color: blue; color:white;}
</style>
<script src="../../js/jquery.js" type="text/javascript"></script>
<script type="text/javascript">
  $(function() {
    $("div").click(function () {
      if($(this).hasClass("toggleStyle")){
        $(this).removeClass("toggleStyle");
      }else{
        $(this).addClass("toggleStyle");
      }
    });
  });
</script></head>
<body>
  <div class="toggleStyle"> hasClass()는 선택된 엘리먼트에
    해당 클래스가 </div>
  <div> 적용되어 있으면 true를 리턴하고</div>
  <div> 아직 적용되어 있지 않으면 false를 리턴한다.</div>
</body>
</html>
```

엘리먼트의 태그 내부의 HTML 콘텐츠 얻어오기

html()

```
$(function() {  
    var htmlStr;  
    $("p").click(function() {  
        htmlStr = $(this).html()  
    });  
    report(htmlStr);  
});  
function report(msg) {  
    $('#console').text(msg);  
}
```



매개변수가 없을 때는 엘리먼트의 HTML 콘텐츠를 가져온다

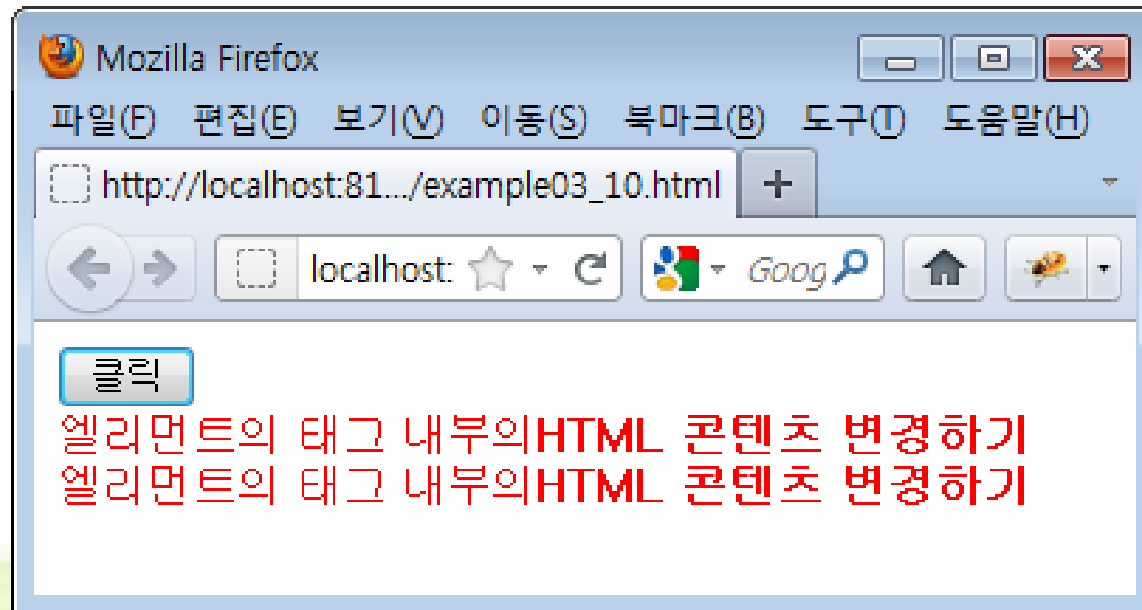
```
<!DOCTYPE html><html><head><meta charset="UTF-8">
<title>엘리먼트의 태그 내부의 HTML 콘텐츠 얻어오기</title>
<style>
p { margin: 14px; cursor: pointer;}
b { text-decoration: underline;}
button { cursor: pointer;}
#console{ font-size: 20px; color: blue; font-weight: bold;}
</style>
<script src="../../js/jquery.js" type="text/javascript"></script>
<script type="text/javascript">
    $(function() {
        var htmlStr;
        $("p").click(function() {
            htmlStr = $(this).html();
            report(htmlStr);
        });
    });
    function report(msg) {
        $('#console').text(msg);
    }
</script></head><body>
    <p>    <b> html()은 </b> 엘리먼트의 <span id="tag"> 내용을 </span>    </p>
    <p>    HTML 형식의<span id="text"> 문자열로 </span> 리턴한다.    </p>
    <p>    버튼을 포함한 <button name="oktbtn">button</button>문장이다.    </p>
    <div id="console"></div>
</body></html>
```


엘리먼트의 태그 내부의 HTML 콘텐츠 변경하기

html(contents)

```
$("#button").click(function() {  
    $("#div").html("<span class='red'> 엘리먼트의 태그 내부의<b>HTML 콘텐  
츠 변경하기</b></span>");  
});
```

인자로 준 **HTML** 코드를 모든 일치되는 엘리먼트의 콘텐츠 설정한다.



```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<meta charset="UTF-8">
```

```
<title>엘리먼트의 태그 내부의 HTML 콘텐츠 변경하기</title>
```

```
<style>
```

```
.red { color: red; }
```

```
</style>
```

```
<script src="../../js/jquery.js" type="text/javascript"></script>
```

```
<script type="text/javascript">
```

```
$(function() {
```

```
    $("button").click(function() {
```

```
        $("div").html("<span class='red'> 엘리먼트의 태그 내부의 "  
            + "<b>HTML 콘텐츠 변경하기</b></span>");
```

```
    });
```

```
});
```

```
</script>
```

```
</head>
```

```
<body>
```

```
<button> 클릭 </button>
```

```
<div></div>
```

```
<div></div>
```

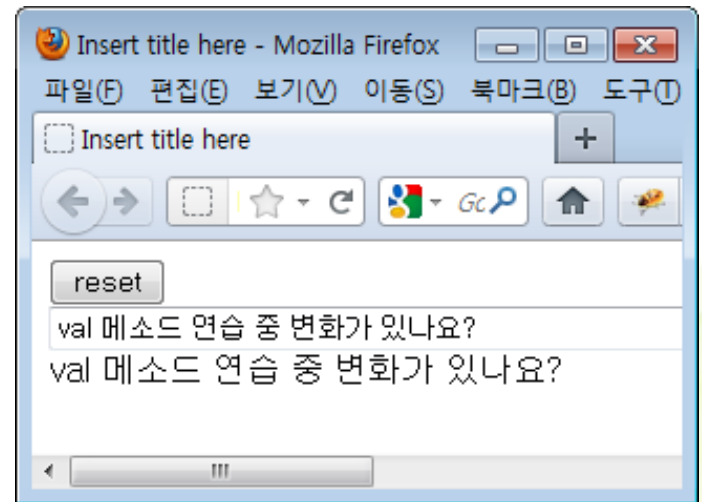
```
</body>
```

```
</html>
```

폼 엘리먼트에서 **VALUE** 어트리뷰트 값 가져오기

val(), val(텍스트)

```
$(function() {  
    $("input").keyup(function () {  
        var value = $(this).val();  
        report(value);  
    }).keyup();  
  
    $("button").click(function() {  
        $("input").val("val 메소드 연습 중")  
    });  
});  
  
function report(msg) {  
    $('#console').text(msg);  
}
```



```
<!DOCTYPE html><html><head>
<meta http-equiv="Content-Type" content="text/html; charset=EUC-KR">
<title>폼 엘리먼트에서 value 어트리뷰트 값 가져오고, 설정하기-val()</title>
<script type="text/javascript" src="../../js/jquery.js"></script>
<script type="text/javascript">
$(function(){
    //이벤트 핸들러 안쪽의 동작 함수는 언제 호출//사용자가 이벤트를 일으켰을 때
    $('input').keyup( function(){
        var value=$(this).val();
        report(value);
    }).keyup(); //문서가 로드되어 준비상태가 되면 개발자가
    //keyup 이벤트를 발생시켜서 핸들러 안쪽의 동작 함수가 한번 실행되도록 한다.
    $('button').click(function(){
        $('input').val("val 메소드 연습 중");
        //개발자가 keyup 이벤트를 발생시켜서
        $('input').keyup();
    });
    function report(msg){
        $('#console').text(msg);
    }
});
</script></head><body>
<button>reset</button>
<input type="text" value="val 메소드 연습 중" size="80">
<div id="console"></div>
</body></html>
```