



Selecting

jQuery 선택자

매우 강력한 jQuery 선택자!

John Resig은 CSS 1-3선택자의 편리함과 신속함을 그대로 jQuery에 도입했습니다.
jQuery가 많은 사람들에게 사랑받는 라이브러리가 된 여러가지 이유중 하나입니다!
즉, 의식개선 B에서 배웠던 CSS1-3 선택자 모두 사용 가능하다는 겁니다.
놀랍지 않습니까?! 초 강력한 jQuery !!!





cssQuery()

getElementsByTagName? Pah!

Introduction

cssQuery() is a powerful *cross-browser* JavaScript function that enables you to use CSS selectors. All CSS1 and CSS2 selectors are allowed plus quite a few CSS3 selectors.

Usage

Syntax

```
elements = cssQuery(selector [, from]);
```

where selector (*required*) is a valid CSS selector and from (*optional*) is a DOM element or a jQuery object representing a set of elements which is filtered by selector.

The function returns a JavaScript array of elements. If there is no match, it returns an empty array.

jQuery 탄생 이전에 이미 딘 에드워드라는 “자바스크립트의 마법사”가 작성한 cssQuery 라이브러리도 있습니다. 대상 선택에 있어 발군의 능력을 발휘합니다만, CSS1-3를 완벽하게 지원하지는 않습니다.





A pure-JavaScript CSS selector engine
designed to be easily dropped in to a host library.



[Documentation](#)

[Github project \(source code\)](#)

[Sizzle discussion group](#)

사실 jQuery의 강력한 선택자는
Sizzle.js 엔진 덕분입니다.
이로 인해 jQuery는 많은
사랑을 받게됩니다.

css

Selector - 셀렉터

CSS 타입(type) 선택자

```
$('h1').addClass('heading1');  
$('div').addClass('division');  
$('a').addClass('anchor link');
```

CSS 그룹핑(grouping)

```
$( 'h1 , div , a' ).addClass( 'heading1 division anchor link' );
```

CSS 자손(descendant) 선택자

```
$('h1 span').addClass('children');
```



CSS 자식(child) 선택자

```
$(‘ul>li’).addClass(‘selectable’);
```

CSS 클래스(class) 선택자

```
$('.lover').addClass('love you');
```

CSS 멀티클래스(multi-class) 선택자

```
$('.lover.me').addClass('love you');
```



CSS 아이디(ID) 선택자

```
$('#aside').removeClass('ad');
```


CSS 유니버설(Universal) 선택자

```
$('#nav *').addClass('theWorld');
```

CSS 인접형제(Sibling) 선택자

```
$('.comment+p').addClass('adjacent');
```

CSS 일반형제(Sibling) 선택자

```
$('.comment~p').addClass('general');
```

CSS 속성(Attribute) 선택자

```
$(p[title]).addClass('title');
```

CSS 속성(Attribute)= 선택자

```
$(`p[title="wow"]`).addClass('title');
```

CSS 속성(Attribute)^{^=} 선택자

```
$('p[title^="wow"]').addClass('title');
```

CSS 속성(Attribute) $\$=$ 선택자

```
 $\$($ 'p[title $\$=$ "wow"]').addClass('title');
```

CSS 속성(Attribute)~= 선택자

```
$(`p[title~=”wow”]`).addClass(“title”);
```


CSS 속성(Attribute)= 선택자

```
$(`p[lang|=`en`]`).addClass(`title`);
```

CSS 속성(Attribute) *= 선택자

```
$('#[title*=language]').addClass('title');
```

CSS 멀티속성(multi-Attribute)[][] 선택자

```
$(`a[href^="http://"][title="external"]`).addClass('title');
```

CSS 가상클래스(Pseudo-class): 선택자

```
$('a:link').addClass('link');  
$('a:visited').addClass('visited');  
$(':hover').addClass('hover');  
$(':active').addClass('active');
```

```
$('input:focus').addClass('focus-in');  
$('#content:lang(en)').addClass('english_text');
```

CSS3 가상클래스(Pseudo-class): 선택자

```
$(':root').addClass('xml-base-root');  
$('li:target').addClass('target-list-item');  
$('#forms input:enabled').addClass('enable');  
$('#forms :disabled').addClass('disable');  
$('#forms :checked').addClass('check');  
$('td:empty').addClass('empty');  
$('#content>.select *':not(span)').removeClass('not-item');
```

CSS3 가상클래스(Pseudo-class): 선택자

```
$(':only-child').addClass('only');  
$('li:first-child').addClass('first-child-list-item');  
$('input:last-child').addClass('last-child-input');  
$('td:nth-child(2n+1)').addClass('stripe');  
$('blockquote cite:nth-last-child(2)').addClass('citation');
```

CSS3 가상클래스(Pseudo-class): 선택자

```
$(':only-of-type').addClass('only');  
$('.focusing p:first-of-type').addClass('focus-in');  
$('span.point:last-of-type').addClass('pointing');  
$('#content p:nth-of-type(3n+2)').addClass('saw');  
$('#footer div:nth-last-child(4)').addClass('fourth');
```

jQuery

Filter - 필터



jQuery != 선택자 필터

```
$(`p[title!=`wow`]`).addClass(`not-include`);
```



jQuery :first 선택자 필터

```
$( 'p:first' ).addClass( 'first' );
```



jQuery :last 선택자 필터

```
$(‘p:last’).addClass(‘last’);
```



jQuery :odd 선택자 필터

```
$( 'p:odd' ).addClass( 'odd' );
```

jQuery :even 선택자 필터

```
$(‘p:even’).addClass(‘even’);
```



jQuery :eq() 선택자 필터

```
$( 'p:eq(0)' ).addClass( 'equal' );
```



jQuery :lt() 선택자 필터

```
$( 'p:lt(3)' ).addClass( 'less-than' );
```

jQuery :gt() 선택자 필터

```
$('p:gt(5)').addClass('greater-than');
```


jQuery :header 선택자 필터

```
$('#content-wrap :header').addClass('headings');
```



jQuery :contains() 선택자 필터

```
$(‘div:contains(“Yamoo”)’).css(‘color’, ‘#30304f’);
```



jQuery :not() 선택자 필터

```
$(‘input:not(:checked)’).addClass(‘disable’);
```

jQuery :has() 선택자 필터

```
$(‘div:has(p)’).addClass(‘has_paragraph’);
```

jQuery :parent 선택자 필터

```
$( 'td:parent' ).addClass( 'has_child' );
```

자식요소(TextNode)를 가진 요소,
반대로는 :empty

jQuery :parent 선택자 필터

```
$( 'td:parent' ).addClass( 'has_child' );
```

Visibility

jQuery - 시각 필터



jQuery :hidden 선택자 필터

```
$('span:hidden').addClass('hidden_element');
```




jQuery :visible 선택자 필터

```
$( 'span:visible' ).addClass( 'visible_element' );
```



jQuery :animated 선택자 필터

```
$(':animated').addClass('animation');
```

Forms

jQuery - 폼

jQuery :input 선택자 필터

```
$('#forms :input').addClass('input');
```



jQuery :text 선택자 필터

```
$('#forms :text').addClass('text');
```



jQuery :password 선택자 필터

```
$('#forms :password').addClass('pass');
```



jQuery :radio 선택자 필터

```
$('#forms :radio').addClass('radio');
```

jQuery :checkbox 선택자 필터

```
$('#forms :checkbox').addClass('checkbox');
```




jQuery :button 선택자 필터

```
$('#forms :button').addClass('button');
```

jQuery :file 선택자 필터

```
$('#forms :file').addClass('file');
```

jQuery :image 선택자 필터

```
$('#forms :image').addClass('img');
```

jQuery :reset 선택자 필터

```
$('#forms :reset').addClass('reset');
```

jQuery :submit 선택자 필터

```
$('#forms :submit').addClass('submit');
```

jQuery :selected 선택자 필터

```
$('#forms :selected').addClass('submit');
```

Original lab by [Karl Swedberg](#) Enhanced by [Cody Lindley](#). FYI (ie6 != supported)

jQuery Selectors

Enter Your Own Selectors (example: li:nth-child(2))

toggle

Basics

`$('code')`

toggle

documentation

`$('#myid')`

toggle

documentation

`$('.myclass')`

toggle

documentation

`$('*')`

toggle

documentation

`$('code, #myid, .myclass')`

toggle

documentation

Hierarchy

`$('div code')`

toggle

documentation

`$('li > ul')`

toggle

documentation

`$('strong + em')`

toggle

documentation

`$('strong ~ em')`

toggle

documentation

Basic Filters

`$('li:first')`

toggle

documentation

`$('li:last')`

toggle

documentation

`$('li:not(li:first)')`

toggle

documentation

Toggle Button = `$('selector').addClass('highlight').animate({ marginLeft: 10}, 'fast');`

[Download This Lab](#)

This sentence is in `<div id="myid">`. It is followed by a horizontal rule.

This is a paragraph, which means it is wrapped in `<p>` and `</p>`. Those "p" tags in the previous sentence are formatted as `<code>`.

- This is the first list item (``) in an unordered list (``).
- This is the second list item. It has a [link](#) in it.
- This is the third list item. It has a `class` of "myclass otherclass"
- This is the fourth list item. It has **strong** text and *emphasized* text.
 - second-level list item 1
 - second-level list item 2

`<p class="myclass">`This is another paragraph. It has class="myclass" Otherwise, nothing **special** about it at all.

This is a paragraph, which means it is wrapped in `<p>` and `</p>`. Those "p" tags in the previous sentence are formatted as `<code>`.

text input

disabled text input

.....

select list w/ options

☒ Radio Checked

☐ Radio Not Checked

☒ Checkbox Checked

☐ Checkbox Not Checked

☐ Another checkbox Not Checked

Submit input

Button input

Reset input

Image input

Button button

찾아보기...

This is a textarea

select list w/ options

another option

a third option

<http://codylindley.com/jqueryselectors/>