

BASIC

All ('*')
Selects all elements
Class ('.class')
Select all elements with the given class
<code>\$('.nameOfClass')</code>
<code>['<div class="nameOfClass">...</div>', '<input class="signup nameOfClass">']</code>
<code>\$('.nameOfClass.otherClass')</code>
<code>['<div class="nameOfClass otherClass">...</div>', '<div class="nameOfClass otherClass thirdClass">...</div>']</code>
Element ('element')
Selects specified element
<code>\$('#div')</code>
<code>['<div id="first">...</div>', '<div id="second">...</div>']</code>
<code>\$('#body')</code>
<code>['<body>...</body>']</code>
ID ('#id')
Selects the one element with the given ID.
<code>\$('#elementOne')</code>
<code>['<div id="elementOne">...</div>']</code>
<code>\$('#uniqueID')</code>
<code>['...']</code>
Multiple ('selector1,selector2,selectorN')
Selects using each of the comma-separated selectors
<code>\$('.classOne,#elementID')</code>
<code>['<div class="classOne">...</div>', '<input id="elementID">']</code>
<code>\$('#form,#uniqueID,.className')</code>
<code>['<form>...</form>', '...', '<ul class="className">...']</code>
ATTRIBUTE
Attribute Contains Prefix ('[name =“value”]')
Selects elements with specified attribute with value followed by a hypen
<code> \$('[formname =“log”]')</code>
<code>['<form formname="log-in">...</form>', '<form formname="log">...</form>']</code>
<code> \$('[callsign="alpha"]')</code>
<code>['<div callsign="alpha-beta">...</div>', '...']</code>

Attribute Contains ('[name*=“value”]')
Selects elements with specified attribute containing value as a substring
<code> \$('[formname*=“log”]')</code>
<code>['<form formname="logged">...</form>', '<form formname="blog">...</form>']</code>
<code> \$('[callsign*=“alpha”]')</code>
<code>['<ul callsign="alphacat">...', '<li callsign="aalphaa">...']</code>
Attribute Contains Word ('[name~=“value”]')
Selects elements with specified attribute containing word (space delimited)
<code> \$('[formname~=“log”]')</code>
<code>['<form formname="my one log">...</form>', '<form formname="hog log dog">...</form>']</code>
<code> \$('[callsign~=“alpha”]')</code>
<code>['<div callsign="alpha beta delta">...</div>', '<h1 callsign="a alpha b">...</h1>']</code>
Attribute Ends With ('[name\$=“value”]')
Selects elements with specified attribute ending with value
<code> \$('[formname\$="log"]')</code>
<code>['<form formname="blog">...</form>', '<form formname="videolog">...</form>']</code>
<code> \$('[callsign\$="alpha"]')</code>
<code>['<div callsign="newalpha">...</div>', '...']</code>
Attribute Equals ('[name=“value”]')
Selects elements with specified attribute matching the value.
<code> \$('[formname="log"]')</code>
<code>['<form formname="log">...</form>', '<div formname="log">...</div>']</code>
<code> \$('[callsign="alpha"]')</code>
<code>['<h1 callsign="alpha">...</h1>', '<div callsign="alpha">...</div>']</code>
Attribute Not Equal ('[name!=“value”]')
Selects elements with specified attributes that don't match value
<code> \$('[formname!="log"]')</code>
<code>['<form formname="register">...</form>', '<form formname="resetpassword">...</form>']</code>
<code> \$('[callsign!="alpha"]')</code>
<code>['<div callsign="tango">...</div>', '...']</code>

Attribute Starts With ('[name^=“value”]')
Selects elements with specified attribute starting with value
<code> \$('[formname^="log"]')</code>
<code>['<form formname="logform">...</form>', '<form formname="loginformation">...</form>']</code>
<code> \$('[callsign^="alpha"]')</code>
<code>['<h1 callsign="alphaone">...</h1>', '<div callsign="alphabravo">...</div>']</code>
Has Attribute ('[name]')
Selects elements with the attribute
<code> \$('[formname]')</code>
<code>['<form formname="signin">...</form>', '<form formname="newuser">...</form>']</code>
<code> \$('[callsign]')</code>
<code>['<div callsign="charlie">...</div>', '...']</code>
Multiple Attribute ('[name=“value”][name2=“value2”]')
Selects elements with the specified attribute selectors
<code> \$('[formname="log"][callsign\$="alpha"]')</code>
<code>['<form formname="log">...</form>', '...']</code>
HIERARCHY
Child ('parent > child')
Selects elements that are direct children of a parent
<code> \$('#form > input')</code>
<code>['<input type="text">', '<input type="password">', '<input type="submit">']</code>
Descendant ('ancestor descendant')
Selects elements that are descendants of an ancestor element
<code> \$('#body li')</code>
<code>['<li id="one">...', '<li id="two">...', '<li id="three">...']</code>
Next Adjacent ('prev + next')
Selects sibling elements that are directly follow the prev selector
<code> \$('#one + li')</code>
<code>['<li id="two">...']</code>
Next Siblings ('prev ~ siblings')
Selects all sibling elements that follow prev selector
<code> \$('#two ~ li')</code>
<code>['<li id="three">...', '<li id="four">...', '<li id="five">...']</code>

BASIC FILTERS

Filters, which begin with colons (:), can be appended to regular selectors to filter out elements that don't match the filter.
For example, \$(':aniimated') will give you all animated on a page, while \$('form:animated') will give you animated elements within a form.
:animated
Selects elements that are currently being animated
:eq()
Selects the element at the given index of the matched set (zero-index)
<code>\$('li:eq(4)')</code>
<code>['<li id="five">...']</code>
<code>\$('li:eq(0)')</code>
<code>['<li id="one">...']</code>
:even
Selects elements with an even index in the matched set (zero-index)
<code>\$('li:even')</code>
<code>['<li id="one">...', '<li id="three">...', '<li id="five">...']</code>
:first
Selects the first matched element
<code>\$('li:first')</code>
<code>['<li id="one">...']</code>
:focus
Selects element that is currently focused
:gt
Selects matched elements with index greater than given index (zero-index)
<code>\$('li:gt(3)')</code>
<code>['<li id="five">...', '<li id="six">...', '<li id="seven">...']</code>
:header
Selects elements that are headers (h1, h2, ...)
<code>\$(':header')</code>
<code>['<h1>...</h1>', '<h3>...</h3>']</code>
:lang()
Selects elements with the given language.
<code>\$(':lang(en-us)')</code>
<code>['<div lang="en-us">...</div>']</code>

:last
Selects the last matched element
<code>\$(':last')</code>
<code>['<li id="last">...']</code>
:lt()
Selects from matched elements with an index less than the given index (zero-index)
<code>\$('li:lt(3)')</code>
<code>['<li id="one">...', '<li id="two">...', '<li id="three">...']</code>
:not()
Selects elements that do not match the given selector
<code>\$('input:not(:password)')</code>
<code>['<input type="text">', '<input type="submit">']</code>
:odd
Selects elements with an odd index in the matched set (zero-index)
<code>\$('li:odd')</code>
<code>['<li id="two">...', '<li id="four">...', '<li id="six">...']</code>
:root
Selects the element at the root of the document
<code>\$(':root')</code>
<code>['<html>...</html>']</code>
:target
Selects the element targetted in URL (e.g. http://url.com/#home)
<code>\$(':target')</code>
<code>['<div id="home">...</div>']</code>
CHILD FILTERS
:first-child
Selects element that is the first child of parent
:first-of-type
Selects first of element type that is a child of parent
:last-child
Selects element that is the last child of parent
:last-of-type
Selects last of element type that is a child of parent

:nth-child()
Selects the child element of a parent matching the given index (1-index)
:nth-last-child()
Selects the child element of parent matching the given index, starting from the last (1-index)
:nth-last-of-type()
Selects the nth element of a type among siblings of same parent, starting from last (1-index)
:nth-of-type()
Selects the nth element of a type among siblings of same parent (1-index)
:only-child
Selects elements when they are only child of their parent
:only-of-type
Selects elements when they have no siblings of same type
CONTENT FILTERS
:contains()
Selects all elements that contain the specified text
<code>\$(':contains("Pretty Printed")')</code>
<code>['<div>Pretty Printed</div>']</code>
:empty
Selects elements with no children or text
<code>\$(':empty')</code>
<code>['<div></div>', '<td></td>']</code>
:has()
Selects elements that have at least one element of given selector
<code>\$('div:has(table)')</code>
<code>['<div><table>...</table></div>']</code>
\$(':parent')
Selects elements that have at least one child element
<code>\$(:parent)</code>
<code>['<table><tr>...</tr></table>','<div><h1>...</h1></div>']</code>

VISIBILITY FILTERS

```
:hidden
```

Selects hidden elements.

```
Hidden elements either have:
display:none;
In a form, type="hidden"
Width and height are 0
```

```
:visible
```

Selects elements that aren't hidden

```
['<input type="image">']
```

Selects elements that are input, select, textarea, and button

```
['<input type="text">', '<textarea>...</textarea>']
```

Selects elements that are type password

```
['<input type="password">']
```

Selects elements that are radio buttons

```
['<input type="radio">']
```

Selects elements that are type reset

```
['<input type="reset">']
```

Selects elements that are selected

```
['<option value="new" selected>...</option>']
```

Selects elements that are type submit

```
['<input type="submit">']
```

Selects elements that are type text

```
['<input type="text">']
```