

# JQUERY - SELECTORS

The jQuery library harnesses the power of Cascading Style Sheets *CSS* selectors to let us quickly and easily access elements or groups of elements in the Document Object Model *DOM*.

A jQuery Selector is a function which makes use of expressions to find out matching elements from a DOM based on the given criteria. Simply you can say, selectors are used to select one or more HTML elements using jQuery. Once an element is selected then we can perform various operations on that selected element.

## The **\$()** factory function

jQuery selectors start with the dollar sign and parentheses – **\$()**. The factory function **\$()** makes use of following three building blocks while selecting elements in a given document –

### S.N. Selector & Description

#### 1 Tag Name

Represents a tag name available in the DOM. For example **\$('p')** selects all paragraphs `<p>` in the document.

#### 2 Tag ID

Represents a tag available with the given ID in the DOM. For example **\$('#some-id')** selects the single element in the document that has an ID of some-id.

#### 3 Tag Class

Represents a tag available with the given class in the DOM. For example **\$('.some-class')** selects all elements in the document that have a class of some-class.

All the above items can be used either on their own or in combination with other selectors. All the jQuery selectors are based on the same principle except some tweaking.

**NOTE** – The factory function **\$()** is a synonym of **jQuery** function. So in case you are using any other JavaScript library where **\$** sign is conflicting with some thing else then you can replace **\$** sign by **jQuery** name and you can use function **jQuery** instead of **\$()**.

## Example

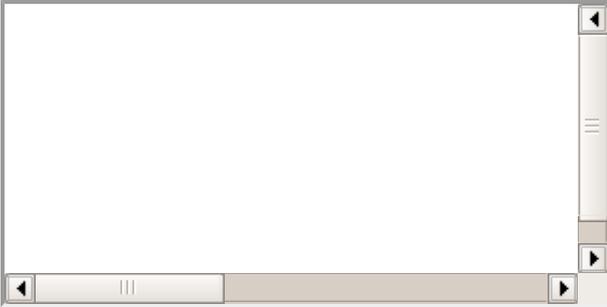
Following is a simple example which makes use of Tag Selector. This would select all the elements with a tag name **p** and will set their background to "yellow".

```
<html>
  <head>
    <title>The jQuery Example</title>
    <script type="text/javascript"
src="http://ajax.googleapis.com/ajax/libs/jquery/2.1.3/jquery.min.js">
    </script>

    <script type="text/javascript" language="javascript">
      $(document).ready(function() {
        $("p").css("background-color", "yellow");
      });
    </script>
  </head>
```

```
<body>
  <div>
    <p >This is a paragraph.</p>
    <p >This is second paragraph.</p>
    <p>This is third paragraph.</p>
  </div>
</body>
</html>
```

This will produce following result –



## How to use Selectors?

The selectors are very useful and would be required at every step while using jQuery. They get the exact element that you want from your HTML document.

Following table lists down few basic selectors and explains them with examples.

S.N.	Selector & Description
1	<b><u>Name</u></b> Selects all elements which match with the given element <b>Name</b> .
2	<b><u>#ID</u></b> Selects a single element which matches with the given <b>ID</b> .
3	<b><u>.Class</u></b> Selects all elements which match with the given <b>Class</b> .
4	<b><u>Universal *</u></b> Selects all elements available in a DOM.
5	<b><u>Multiple Elements E, F, G</u></b> Selects the combined results of all the specified selectors <b>E, F</b> or <b>G</b> .

## Selectors Examples

Similar to above syntax and examples, following examples would give you understanding on using different type of other useful selectors –

S.N.	Selector & Description
------	------------------------

- 1     **\$(\*)**  
This selector selects all elements in the document.
- 2     **\$("p > \*")**  
This selector selects all elements that are children of a paragraph element.
- 3     **\$("#specialID")**  
This selector function gets the element with .
- 4     **\$(".specialClass")**  
This selector gets all the elements that have the class of *specialClass*.
- 5     **\$("li:not(.myclass)")**  
Selects all elements matched by <li> that do not have .
- 6     **\$("a#specialID.specialClass")**  
This selector matches links with an id of *specialID* and a class of *specialClass*.
- 7     **\$("p a.specialClass")**  
This selector matches links with a class of *specialClass* declared within <p> elements.
- 8     **\$("ul li:first")**  
This selector gets only the first <li> element of the <ul>.
- 9     **\$("#container p")**  
Selects all elements matched by <p> that are descendants of an element that has an id of *container*.
- 10    **\$("li > ul")**  
Selects all elements matched by <ul> that are children of an element matched by <li>
- 11    **\$("strong + em")**  
Selects all elements matched by <em> that immediately follow a sibling element matched by <strong>.
- 12    **\$("p ~ ul")**  
Selects all elements matched by <ul> that follow a sibling element matched by <p>.
- 13    **\$("code, em, strong")**  
Selects all elements matched by <code> or <em> or <strong>.
- 14    **\$("p strong, .myclass")**  
Selects all elements matched by <strong> that are descendants of an element matched

by `<p>` as well as all elements that have a class of *myclass*.

15 **`$(":empty")`**

Selects all elements that have no children.

16 **`$("p:empty")`**

Selects all elements matched by `<p>` that have no children.

17 **`$("div[p]")`**

Selects all elements matched by `<div>` that contain an element matched by `<p>`.

18 **`$("p[.myclass]")`**

Selects all elements matched by `<p>` that contain an element with a class of *myclass*.

19 **`$("a[@rel]")`**

Selects all elements matched by `<a>` that have a `rel` attribute.

20 **`$("input[@name=myname]")`**

Selects all elements matched by `<input>` that have a `name` value exactly equal to *myname*.

21 **`$("input[@name^=myname]")`**

Selects all elements matched by `<input>` that have a `name` value beginning with *myname*.

22 **`(" a[@rel=self]")`**

Selects all elements matched by `<a>` that have **rel** attribute value ending with *self*.

23 **`$("a[@href*=domain.com]")`**

Selects all elements matched by `<a>` that have an `href` value containing `domain.com`.

24 **`$("li:even")`**

Selects all elements matched by `<li>` that have an even index value.

25 **`$("tr:odd")`**

Selects all elements matched by `<tr>` that have an odd index value.

26 **`$("li:first")`**

Selects the first `<li>` element.

27 **`$("li:last")`**

Selects the last `<li>` element.

28	<b><code>\$("li:visible")</code></b>	Selects all elements matched by <code>&lt;li&gt;</code> that are visible.
29	<b><code>\$("li:hidden")</code></b>	Selects all elements matched by <code>&lt;li&gt;</code> that are hidden.
30	<b><code>\$(":radio")</code></b>	Selects all radio buttons in the form.
31	<b><code>\$(":checked")</code></b>	Selects all checked boxes in the form.
32	<b><code>\$(":input")</code></b>	Selects only form elements <i>input, select, textarea, button</i> .
33	<b><code>\$(":text")</code></b>	Selects only text elements <i>input[type = text]</i> .
34	<b><code>\$("li:eq(2)")</code></b>	Selects the third <code>&lt;li&gt;</code> element.
35	<b><code>\$("li:eq(4)")</code></b>	Selects the fifth <code>&lt;li&gt;</code> element.
36	<b><code>\$("li:lt(2)")</code></b>	Selects all elements matched by <code>&lt;li&gt;</code> element before the third one; in other words, the first two <code>&lt;li&gt;</code> elements.
37	<b><code>\$("p:lt(3)")</code></b>	selects all elements matched by <code>&lt;p&gt;</code> elements before the fourth one; in other words the first three <code>&lt;p&gt;</code> elements.
38	<b><code>\$("li:gt(1)")</code></b>	Selects all elements matched by <code>&lt;li&gt;</code> after the second one.
39	<b><code>\$("p:gt(2)")</code></b>	Selects all elements matched by <code>&lt;p&gt;</code> after the third one.
40	<b><code>\$("div/p")</code></b>	Selects all elements matched by <code>&lt;p&gt;</code> that are children of an element matched by <code>&lt;div&gt;</code> .
41	<b><code>\$("div//code")</code></b>	Selects all elements matched by <code>&lt;code&gt;</code> that are descendants of an element matched by

<div>.

42 **`$("//p//a")`**

Selects all elements matched by <a> that are descendants of an element matched by <p>

43 **`$("li:first-child")`**

Selects all elements matched by <li> that are the first child of their parent.

44 **`$("li:last-child")`**

Selects all elements matched by <li> that are the last child of their parent.

45 **`$(":parent")`**

Selects all elements that are the parent of another element, including text.

46 **`$("li:contains(second)")`**

Selects all elements matched by <li> that contain the text second.

You can use all the above selectors with any HTML/XML element in generic way. For example if selector **`$("li:first")`** works for <li> element then **`$("p:first")`** would also work for <p> element.

Loading [MathJax]/jax/output/HTML-CSS/jax.js