

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 <code>\$('p')</code> selects all paragraphs <code><p></code> in the document. |
| 2 | Tag ID Represents a tag available with the given ID in the DOM. For example <code>\$('#some-id')</code> 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 <code>\$('.some-class')</code> 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 class = "myclass">This is a paragraph.</p>
    <p id = "myid">This is second paragraph.</p>
    <p>This is third paragraph.</p>
  </div>
</body>
</html>
```

This will produce following result –

This is a paragraph.
 This is second paragraph.
 This is third paragraph.

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

JQUERY - ELEMENT NAME SELECTOR

Description

The element selector selects all the elements that have a tag name of T.

Syntax

Here is the simple syntax to use this selector –

```
$('tagname')
```

Parameters

Here is the description of all the parameters used by this selector –

- **tagname** – Any standard HTML tag name like div, p, em, img, li etc.

Returns

Like any other jQuery selector, this selector also returns an array filled with the found elements.

Example

- **\$(**p**)** – Selects all elements with a tag name of **p** in the document.
- **\$(**div**)** – Selects all elements with a tag name of **div** in the document.

Following example would select all the divisions and will apply yellow color to their background –

```
<html>

<head>
  <title>The Selector 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() {
      /* This would select all the divisions */
      $("div").css("background-color", "yellow");
    });
  </script>
</head>
</html>
```

```
});  
</script>  
</head>  
  
<body>  
  
<div class = "big" id = "div1">  
  <p>This is first division of the DOM.</p>  
</div>  
  
<div class = "medium" id = "div2">  
  <p>This is second division of the DOM.</p>  
</div>  
  
<div class = "small" id = "div3">  
  <p>This is third division of the DOM</p>  
</div>  
  
</body>  
  
</html>
```

This will produce following result –

This is first division of the DOM.

This is second division of the DOM.

This is third division of the DOM

JQUERY - ELEMENT ID SELECTOR

Description

The element ID selector selects a single element with the given id attribute.

Syntax

Here is the simple syntax to use this selector –

```
$('#elementid')
```

Parameters

Here is the description of all the parameters used by this selector –

- **elementid** – This would be an element ID. If the id contains any special characters like periods or colons you have to escape those characters with backslashes.

Returns

Like any other jQuery selector, this selector also returns an array filled with the found element.

Example

- `$('#myid')` – Selects a single element with the given id myid.
- `$('#div#yourid')` – Selects a single division with the given id yourid.

Following example would select second division and will apply yellow color to its background –

```
<html>

<head>
  <title>The Selector 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() {
    /* This would select second division only*/
    $("#div2").css("background-color", "yellow");
  });
</script>
</head>
<body>

<div class = "big" id = "div1">
  <p>This is first division of the DOM.</p>
</div>

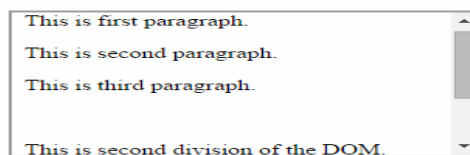
<div class = "medium" id = "div2">
  <p>This is second division of the DOM.</p>
</div>

<div class = "small" id = "div3">
  <p>This is third division of the DOM</p>
</div>

</body>

</html>
```

This will produce following result –



JQUERY - ELEMENT CLASS SELECTOR

Description

The element class selector selects all the elements which match with the given class of the elements.

Syntax

Here is the simple syntax to use this selector –

```
$('.classid')
```

Parameters

Here is the description of all the parameters used by this selector –

- **classid** – This is class ID available in the document.

Returns

Like any other jQuery selector, this selector also returns an array filled with the found elements.

Example

- **\$('.big')** – Selects all the elements with the given class ID **big**.
- **\$('.p.small')** – Selects all the paragraphs with the given class ID **small**.
- **\$('.big.small')** – Selects all the elements with a class of **big** and **small**.

Following example would select all divisions with class **.big** and will apply yellow color to its background –

```
<html>

<head>
  <title>The Selector 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() {  
    /* This would select second division only*/  
    $(".big").css("background-color", "yellow");  
});  
</script>  
</head>  
  
<body>  
  
    <div class = "big" id="div1">  
        <p>This is first division of the DOM.</p>  
    </div>  
  
    <div class = "medium" id = "div2">  
        <p>This is second division of the DOM.</p>  
    </div>  
  
    <div class = "small" id = "div3">  
        <p>This is third division of the DOM</p>  
    </div>  
  
</body>  
  
</html>
```

This will produce following result –

This is first division of the DOM.
This is second division of the DOM.
This is third division of the DOM

JQUERY - UNIVERSAL SELECTOR

Description

The universal selector selects all the elements available in the document.

Syntax

Here is the simple syntax to use this selector –

```
$( '* ' )
```

Parameters

Here is the description of all the parameters used by this selector –

- * – A symbolic star.

Returns

Like any other jQuery selector, this selector also returns an array filled with the found elements.

Example

- \$('* ') selects all the elements available in the document.

Following example would select all the elements and will apply yellow color to their background. Try to understand that this selector will select every element including head, body etc.

```
<html>
<head>
  <title>The Selector 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() {
      /* This would select all the elements */
      $( " * " ).css("background-color", "yellow");
    });
```

```
</script>

</head>

<body>

  <div class = "big" id = "div1">
    <p>This is first division of the DOM.</p>
  </div>

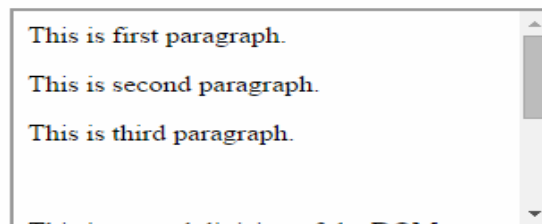
  <div class = "medium" id = "div2">
    <p>This is second division of the DOM.</p>
  </div>

  <div class = "small" id = "div3">
    <p>This is third division of the DOM</p>
  </div>

</body>

</html>
```

This will produce following result –



JQUERY - MULTIPLE ELEMENTS SELECTOR

Description

This Multiple Elements selector selects the combined results of all the specified selectors E, F or G.

You can specify any number of selectors to combine into a single result. Here order of the DOM elements in the jQuery object aren't necessarily identical.

Syntax

Here is the simple syntax to use this selector –

```
$( 'E, F, G,....' )
```

Parameters

Here is the description of all the parameters used by this selector –

- **E** – Any valid selector
- **F** – Any valid selector
- **G** – Any valid selector
-

Returns

Like any other jQuery selector, this selector also returns an array filled with the found elements.

Example

- `$('div, p')` – selects all the elements matched by **div** or **p**.
- `$('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**.
- `$('p strong, #myid')` – selects a single elements matched by **strong** that is descendant of an element matched by **p** as well as element whose id is **myid**.

Following example would select elements with class ID **big** and element with ID **div3** and will apply yellow color to its background –

```
<html>
<head>
<title>The Selector 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() {
        $(".big, #div3").css("background-color", "yellow");
    });
</script>
</head>

<body>

    <div class = "big" id = "div1">
        <p>This is first division of the DOM.</p>
    </div>

    <div class = "medium" id = "div2">
        <p>This is second division of the DOM.</p>
    </div>

    <div class = "small" id = "div3">
        <p>This is third division of the DOM</p>
    </div>

</body>

</html>
```

This will produce following result –

This is first division of the DOM.

This is second division of the DOM.

This is third division of the DOM

Selectors Examples

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

Here, you have different type of other useful selectors –

| S.N. | Selector & Description |
|------|---|
| 1 | <p><code>\$('*')</code></p> <p>This selector selects all elements in the document.</p> |
| 2 | <p><code>\$("p > *")</code></p> <p>This selector selects all elements that are children of a paragraph element.</p> |
| 3 | <p><code>\$("#specialID")</code></p> <p>This selector function gets the element with <code>id="specialID"</code>.</p> |
| 4 | <p><code>\$(".specialClass")</code></p> <p>This selector gets all the elements that have the class of <i>specialClass</i>.</p> |
| 5 | <p><code>\$("li:not(.myclass)")</code></p> <p>Selects all elements matched by <code></code> that do not have class = "myclass".</p> |

| | |
|----|--|
| 6 | <code>\$("a#specialID.specialClass")</code> This selector matches links with an id of <i>specialID</i> and a class of <i>specialClass</i> . |
| 7 | <code>\$("p a.specialClass")</code> This selector matches links with a class of <i>specialClass</i> declared within <code><p></code> elements. |
| 8 | <code>\$("ul li:first")</code> This selector gets only the first <code></code> element of the <code></code> . |
| 9 | <code>\$("#container p")</code> Selects all elements matched by <code><p></code> that are descendants of an element that has an id of <i>container</i> . |
| 10 | <code>\$("li > ul")</code> Selects all elements matched by <code></code> that are children of an element matched by <code></code> |
| 11 | <code>\$("strong + em")</code> Selects all elements matched by <code></code> that immediately follow a sibling element matched by <code></code> . |
| 12 | <code>\$("p ~ ul")</code> Selects all elements matched by <code></code> that follow a sibling element matched by <code><p></code> . |
| 13 | <code>\$("code, em, strong")</code> Selects all elements matched by <code><code></code> or <code></code> or <code></code> . |
| 14 | <code>\$("p strong, .myclass")</code> Selects all elements matched by <code></code> that are descendants of an element matched by <code><p></code> as well as all elements that have a class of <i>myclass</i> . |
| 15 | <code>\$(":empty")</code> Selects all elements that have no children. |

| | |
|----|---|
| 16 | <code>\$("p:empty")</code> Selects all elements matched by <code><p></code> that have no children. |
| 17 | <code>\$("div[p]")</code> Selects all elements matched by <code><div></code> that contain an element matched by <code><p></code> . |
| 18 | <code>\$("p[.myclass]")</code> Selects all elements matched by <code><p></code> that contain an element with a class of <i>myclass</i> . |
| 19 | <code>\$("a[@rel]")</code> Selects all elements matched by <code><a></code> that have a <code>rel</code> attribute. |
| 20 | <code>\$("input[@name = myname]")</code> Selects all elements matched by <code><input></code> that have a <code>name</code> value exactly equal to <i>myname</i> . |
| 21 | <code>\$("input[@name^=myname]")</code> Selects all elements matched by <code><input></code> that have a <code>name</code> value beginning with <i>myname</i> . |
| 22 | <code>("a[@rel("a[@rel=self]")]</code> Selects all elements matched by <code><a></code> that have rel attribute value ending with <i>self</i> . |
| 23 | <code>\$("a[@href*=domain.com]")</code> Selects all elements matched by <code><a></code> that have an <code>href</code> value containing <code>domain.com</code> . |
| 24 | <code>\$("li:even")</code> Selects all elements matched by <code></code> that have an even index value. |
| 25 | <code>\$("tr:odd")</code> Selects all elements matched by <code><tr></code> that have an odd index value. |
| 26 | <code>\$("li:first")</code> Selects the first <code></code> element. |

| | |
|----|--|
| 27 | \$("li:last") Selects the last element. |
| 28 | \$("li:visible") Selects all elements matched by that are visible. |
| 29 | \$("li:hidden") Selects all elements matched by that are hidden. |
| 30 | \$(":radio") Selects all radio buttons in the form. |
| 31 | \$(":checked") Selects all checked box in the form. |
| 32 | \$(":input") Selects only form elements input,select,textarea,buttoninput,select,textarea,button. |
| 33 | \$(":text") Selects only text elements input[type=text]input[type=text]. |
| 34 | \$("li:eq(2)") Selects the third element. |
| 35 | \$("li:eq(4)") Selects the fifth element. |
| 36 | \$("li:lt(2)") Selects all elements matched by element before the third one; in other words, the first two elements. |
| 37 | \$("p:lt(3)") selects all elements matched by <p> elements before the fourth one; in other words the first three <p> elements. |

| | |
|----|---|
| 38 | <code>\$("li:gt(1)")</code> Selects all elements matched by <code></code> after the second one. |
| 39 | <code>\$("p:gt(2)")</code> Selects all elements matched by <code><p></code> after the third one. |
| 40 | <code>\$("div/p")</code> Selects all elements matched by <code><p></code> that are children of an element matched by <code><div></code> . |
| 41 | <code>\$("div//code")</code> Selects all elements matched by <code><code></code> that are descendants of an element matched by <code><div></code> . |
| 42 | <code>\$("//p//a")</code> Selects all elements matched by <code><a></code> that are descendants of an element matched by <code><p></code> |
| 43 | <code>\$("li:first-child")</code> Selects all elements matched by <code></code> that are the first child of their parent. |
| 44 | <code>\$("li:last-child")</code> Selects all elements matched by <code></code> that are the last child of their parent. |
| 45 | <code>\$(":parent")</code> Selects all elements that are the parent of another element, including text. |
| 46 | <code>\$("li:contains(second)")</code> Selects all elements matched by <code></code> that contain the text second. |

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