

# jQuery Selectors

## Selecting Elements with jQuery

JavaScript is most commonly used to get or modify the content or value of the HTML elements on the page, as well as to apply some effects like show, hide, animations etc. But, before you can perform any action you need to find or select the target HTML element.

Selecting the elements through a typical JavaScript approach could be very painful, but the jQuery works like a magic here. The ability of making the DOM elements selection simple and easy is one of the most powerful feature of the jQuery.

**Tip:** The jQuery supports almost all the **selectors** defined in the latest CSS3 specifications, as well as it has its own custom selectors. These custom selectors greatly enhance the capabilities selecting the HTML elements on a page.

In the following sections, you will see some of the common ways of selecting the elements on a page and do something with them using the jQuery.

## Selecting Elements by ID

You can use the ID selector to select a single element with the unique ID on the page.

For example, the following jQuery code will select and highlight an element having the ID attribute `id="mark"`, when the document is ready to be manipulated.

```
<script> $(document).ready(function() {  
  
// Highlight element with id mark $("#mark").css("background",  
"yellow"); }); </script>
```

In the example above, the `$(document).ready()` is an event that is used to manipulate a page safely with the jQuery. Code included inside this event will only run once the page DOM is ready. We'll learn more about the events in next chapter.

---

## Selecting Elements by Class Name

The class selector can be used to select the elements with a specific class.

For example, the following jQuery code will select and highlight the elements having the class attribute `class="mark"`, when the document is ready.

```
<script> $(document).ready(function() {  
  
// Highlight elements with class mark  
$(".mark").css("background", "yellow"); }); </script>
```

## Selecting Elements by Name

The element selector can be used to select elements based on the element name.

For example, the following jQuery code will select and highlight all the paragraph i.e. the `<p>` elements of the document when it is ready.

```
<script> $(document).ready(function() {
```

```
// Highlight paragraph elements $("p").css("background",  
"yellow"); }); </script>
```

---

## Selecting Elements by Attribute

You can use the attribute selector to select an element by one of its HTML attributes, such as a link's `target` attribute or an input's `type` attribute, etc.

For example, the following jQuery code will select and highlight all the text inputs i.e. `<input>` elements with the `type="text"`, when the document is ready.

```
<script> $(document).ready(function() {  
  
// Highlight paragraph elements  
$('input[type="text"]').css("background", "yellow"); });  
</script>
```

---

## Selecting Elements by Compound CSS Selector

You can also combine the CSS selectors to make your selection even more precise.

For instance, you can combine the class selector with an element selector to find the elements in a document that has certain type and class.

```
<script> $(document).ready(function() {  
  
// Highlight only paragraph elements with class mark  
$("p.mark").css("background", "yellow");
```

```
// Highlight only span elements inside the element with ID mark
$("#mark span").css("background", "yellow");

// Highlight li elements inside the ul elements $("ul
li").css("background", "red");

// Highlight li elements only inside the ul element with id mark
$("ul#mark li").css("background", "yellow");

// Highlight li elements inside all the ul element with class
mark $("ul.mark li").css("background", "green");

// Highlight all anchor elements with target blank
$('a[target="_blank"]').css("background", "yellow"); });
</script>
```

---

## jQuery Custom Selector

In addition to the [CSS defined selectors](#), jQuery provides its own custom selector to further enhancing the capabilities of selecting elements on a page.

```
<script> $(document).ready(function(){

// Highlight table rows appearing at odd places
$("tr:odd").css("background", "yellow");

// Highlight table rows appearing at even places
$("tr:even").css("background", "orange");

// Highlight first paragraph element
$("p:first").css("background", "red");
```

```
// Highlight last paragraph element
$("p:last").css("background", "green");

// Highlight all input elements with type text inside a form
$("form :text").css("background", "purple");

// Highlight all input elements with type password inside a form
$("form :password").css("background", "blue");

// Highlight all input elements with type submit inside a form
$("form :submit").css("background", "violet"); }); </script>
```