

Basic syntax

https://www.w3schools.com/jquery/jquery_ref_overview.asp

`$(this).hide()` - hides the current element.

`$("p").hide()` - hides all <p> elements.

`$(".test").hide()` - hides all elements with class="test".

`$("#test").hide()` - hides the element with id="test".

`$('td[name="tcol1"]')` // matches exactly 'tcol1'

`$('td[name^="tcol"]')` // matches those that begin with 'tcol'

`$('td[name$="tcol"]')` // matches those that end with 'tcol'

`$('td[name*="tcol"]')` // matches those that contain 'tcol'

The Document Ready Event

You might have noticed that all jQuery methods in our examples, are inside a document ready event:

```
$(document).ready(function() {  
  
    // jQuery methods go here...  
  
});
```

All selectors

<https://www.w3schools.com/jquery/tryssel.asp>

`$("#Lastname")`

`$(".intro")`

`$(".intro, #Lastname")`

`$("h1")`

`$("h1, p")`

\$("p:first")
\$("p:last")
\$("tr:even")
\$("tr:odd")
\$("p:first-child")
\$("p:first-of-type")
\$("p:last-child")
\$("p:last-of-type")
\$("li:nth-child(1)")
\$("li:nth-last-child(1)")
\$("li:nth-of-type(2)")
\$("li:nth-last-of-type(2)")
\$("b:only-child")
\$("h3:only-of-type")
\$("div > p")
\$("div p")
\$("ul + p")
\$("ul ~ table")
\$("ul li:eq(0)")
\$("ul li:gt(0)")
\$("ul li:lt(2)")
\$(":header")
\$(":header:not(h1)")
\$(":animated")
\$(":focus")
\$(":contains(Duck)")
\$("div:has(p)")
\$(":empty")
\$(":parent")
\$("p:hidden")
\$("table:visible")
\$(":root")
\$("p:lang(it)")
\$("[id]")
\$("[id=my-Address]")
\$("p[id!=my-Address]")
\$("[id\$=ess]")
\$("[id|=my]")
\$("[id^=L]")
\$("[title~=beautiful]")
\$("[id*=s]")
\$(":input")
\$(":text")
\$(":password")
\$(":radio")
\$(":checkbox")
\$(":submit")

```
$(":reset")
$(":button")
$(":image")
$(":file")
$(":enabled")
$(":disabled")
$(":selected")
$(":checked")
$("*")
```

Events

Mouse Events	Keyboard Events	Form Events	Document/Window Events
click	keypress	submit	load
dblclick	keydown	change	resize
mouseenter	keyup	focus	scroll
mouseleave		blur	unload

```
$("p").click(function() {
    $(this).hide();
});
$("p").dblclick(function() {
    $(this).hide();
});
$("#p1").mouseenter(function() {
    alert("You entered p1!");
});
$("#p1").mouseleave(function() {
    alert("Bye! You now leave p1!");
});
$("#p1").mousedown(function() {
    alert("Mouse down over p1!");
});
$("#p1").mouseup(function() {
    alert("Mouse up over p1!");
});
```

The `hover()` method takes two functions and is a combination of the `mouseenter()` and `mouseleave()` methods.

```
$("#p1").hover(function() {
    alert("You entered p1!");
},
```

```
function() {
    alert("Bye! You now leave p1!");
};
```

The `focus()` method attaches an event handler function to an HTML form field.

```
$("#input").focus(function() {
    $(this).css("background-color", "#cccccc");
});
```

The `blur()` method attaches an event handler function to an HTML form field.

```
$("#input").blur(function() {
    $(this).css("background-color", "#ffffff");
});
```

The `on()` method attaches one or more event handlers for the selected elements.

```
$("#p").on("click", function() { // one event
    $(this).hide();
});
$("#p").on({ // multiple events
    mouseenter: function() {
        $(this).css("background-color", "lightgray");
    },
    mouseleave: function() {
        $(this).css("background-color", "lightblue");
    },
    click: function() {
        $(this).css("background-color", "yellow");
    }
});
```

Effects - hide/show/toggle

```
$("#button").click(function() {
    $("#p").hide(1000); // "slow", "fast", 1000 milliseconds
});
$("#button").click(function() {
    $("#p").show(1000);
});
$("#button").click(function() {
    $("#p").toggle();
});
```

Effects - Fading

```
$("#button").click(function() {
    $("#div1").fadeIn();
    $("#div2").fadeIn("slow");
});
```

```

    $("#div3").fadeIn(3000);
});

$("#button").click(function() {
    $("#div1").fadeOut();
    $("#div2").fadeOut("slow");
    $("#div3").fadeOut(3000);
});

$("#button").click(function() {
    $("#div1").fadeToggle();
    $("#div2").fadeToggle("slow");
    $("#div3").fadeToggle(3000);
});

// fadeTo() method specifies fading to a given opacity (value between 0
and 1)
$("#button").click(function() {
    $("#div1").fadeTo("slow", 0.15);
    $("#div2").fadeTo("slow", 0.4);
    $("#div3").fadeTo("slow", 0.7);
});

```

Effects - Slide

```

$("#flip").click(function() {
    $("#panel").slideDown();
});

$("#flip").click(function() {
    $("#panel").slideUp();
});

$("#flip").click(function() {
    $("#panel").slideToggle();
});

```

Effects - Animation

```

$("#button").click(function() {
    var div = $("#div");
    div.animate({height: '300px', opacity: '0.4'}, "slow");
    div.animate({width: '300px', opacity: '0.8'}, "slow");
    div.animate({height: '100px', opacity: '0.4'}, "slow");
    div.animate({width: '100px', opacity: '0.8'}, "slow");
});

$("#button").click(function() {
    var div = $("#div");
    div.animate({left: '100px'}, "slow");
    div.animate({fontSize: '3em'}, "slow");
});

```

Effects - Stop

```
$("#stop").click(function() {  
    $("#panel").stop();  
});
```

Effects - callback

```
$("#button").click(function() {  
    $("#p").hide("slow", function() {  
        alert("The paragraph is now hidden");  
    });  
});
```

Method chaining

```
$("#p1").css("color", "red").slideUp(2000).slideDown(2000);
```

jQuery HTML

Get Content - text(), html(), and val()

- **text()** - Sets or returns the text content of selected elements
- **html()** - Sets or returns the content of selected elements (including HTML markup)
- **val()** - Sets or returns the value of form fields

```
$("#btn1").click(function() {  
    alert("Text: " + $("#test").text());  
});  
$("#btn2").click(function() {  
    alert("HTML: " + $("#test").html());  
});
```

Get Attributes - attr()

attr() method is used to get attribute values

```
$("#button").click(function() {  
    alert($("#w3s").attr("href"));  
});
```

Set Content - text(), html(), and val()

```
$(document).ready(function() {  
    $("#btn1").click(function() {  
        $("#test1").text("Hello world!");  
    });  
    $("#btn2").click(function() {
```

```

        $("#test2").html("<b>Hello world!</b>");
    });
    $("#btn3").click(function(){
        $("#test3").val("Dolly Duck");
    });
    $("#btn4").click(function(){
        $("#test1").text("This is a paragraph.");
        $("#test2").html("This is another paragraph.");
        $("#test3").val("Mickey Mouse");
    });
});

$("#btn1").click(function(){
    $("#test1").text(function(i, origText){
        return "Old text: " + origText + " New text: Hello world!
        (index: " + i + ")";
    });
});
$("#btn2").click(function(){
    $("#test2").html(function(i, origText){
        return "Old html: " + origText + " New html: Hello
<b>world!</b>
        (index: " + i + ")";
    });
});

$("#button").click(function(){
    $("#w3s").attr("href", "https://www.w3schools.com/jquery/");
});
$("#button").click(function(){
    $("#w3s").attr("href", function(i, origValue){
        return origValue + "/jquery/";
    });
});

```

Add Elements

```

$("#p").append("Some appended text.");
$("#p").prepend("Some prepended text.");
function appendText() {
    var txt1 = "<p>Text.</p>";    // Create element with HTML
    var txt2 = $("<p></p>").text("Text.");    // Create with jQuery
    var txt3 = document.createElement("p");    // Create with DOM
    txt3.innerHTML = "Text.";
    $("#body").append(txt1, txt2, txt3);    // Append elements
}
$("#img").after("Some text after"); //add text before/after image.

```

```
$("#img").before("Some text before");
```

Remove Elements

```
$("#div1").remove();  
$("#div1").empty();  
$("p").remove(".test"); # jQuery selector syntaxes  
$("p").remove(".test, .demo");
```

HTML Reference

https://www.w3schools.com/jquery/jquery_ref_html.asp

jQuery CSS

addClass()

```
$("#button").click(function() {  
    $("h1, h2, p").addClass("blue");  
    $("div").addClass("important blue");  
});
```

removeClass()

```
$("#button").click(function() {  
    $("h1, h2, p").removeClass("blue");  
});
```

toggleClass()

```
$("#button").click(function() {  
    $("h1, h2, p").toggleClass("blue");  
});
```

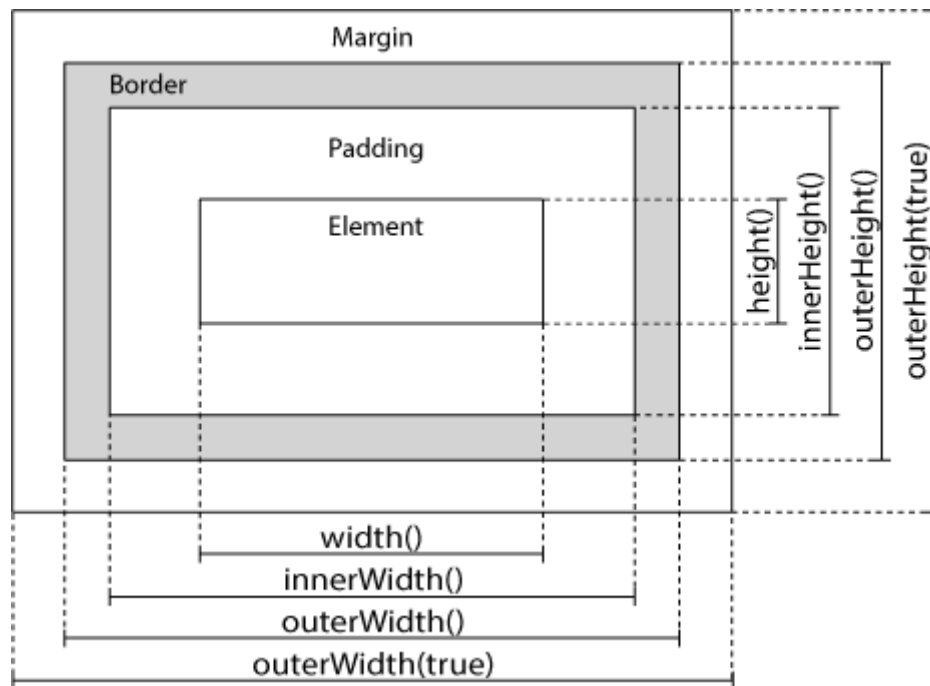
CSS Reference

https://www.w3schools.com/jquery/jquery_ref_html.asp

CSS methods

```
//get css property  
$("p").css("background-color");  
// set css property  
$("p").css("background-color", "yellow");  
// set multiple css properties  
$("p").css({"background-color": "yellow", "font-size": "200%"});
```


Dimensions



Dimension methods

- `width()`
- `height()`
- `innerWidth()`
- `innerHeight()`
- `outerWidth()`
- `outerHeight()`

```
$("button").click(function(){  
    var txt = "";  
    txt += "Width: " + $("#div1").width() + "<br>";  
    txt += "Height: " + $("#div1").height();  
    $("#div1").html(txt);  
});
```

```
$("button").click(function(){  
    var txt = "";  
    txt += "Inner width: " + $("#div1").innerWidth() + "<br>";  
    txt += "Inner height: " + $("#div1").innerHeight();  
    $("#div1").html(txt);  
});
```

```
$("button").click(function(){  
    var txt = "";  
    txt += "Outer width: " + $("#div1").outerWidth() + "<br>";
```

```

    txt += "Outer height: " + $("#div1").outerHeight();
    $("#div1").html(txt);
});

$("#button").click(function() {
    var txt = "";
    txt += "Outer width (+margin): " + $("#div1").outerWidth(true) +
"</br>";
    txt += "Outer height (+margin): " +
$("#div1").outerHeight(true);
    $("#div1").html(txt);
});

$("#button").click(function() {
    var txt = "";
    txt += "Document width/height: " + $(document).width();
    txt += "x" + $(document).height() + "\n";
    txt += "Window width/height: " + $(window).width();
    txt += "x" + $(window).height();
    alert(txt);
});

$("#button").click(function() {
    $("#div1").width(500).height(500);
});

```

jQuery Traversing

Reference

https://www.w3schools.com/jquery/jquery_ref_traversing.asp

Ancestors

// parents() returns all ancestors of all **** elements

```

$(document).ready(function() {
    $("span").parents();
});

```

// returns all ancestor elements between a **** and a **<div>** element

```

$(document).ready(function() {
    $("span").parentsUntil("div");
});

```

```
});
```

Descendants

// returns all elements that are direct children of each `<div>` elements

```
$(document).ready(function() {  
    $("div").children();  
});
```

// returns all `<p>` elements with the class name "first", that are direct children of `<div>`

```
$(document).ready(function() {  
    $("div").children("p.first");  
});
```

// returns all `` elements that are descendants of `<div>`

```
$(document).ready(function() {  
    $("div").find("span");  
});  
$(document).ready(function() {  
    $("div").find("*");  
});
```

Siblings

// returns all sibling elements of `<h2>`

```
$(document).ready(function() {  
    $("h2").siblings();  
});
```

// returns all sibling elements of `<h2>` that are `<p>` elements

```
$(document).ready(function() {  
    $("h2").siblings("p");  
});
```

// returns the next sibling of `<h2>`

```
$(document).ready(function() {  
    $("h2").next();  
});
```

// returns all next sibling elements of `<h2>`

```
$(document).ready(function() {  
    $("h2").nextAll();  
});
```

// returns all sibling elements between a `<h2>` and a `<h6>` element

```
$(document).ready(function() {  
    $("h2").nextUntil("h6");  
});
```

Filtering

```
// first() method returns the first element of the specified elements
$(document).ready(function() {
    $("div").first();
});

// last() method returns the last element of the specified elements.
$(document).ready(function() {
    $("div").last();
});

// eq() method returns an element with a specific index number of the
selected elements
$(document).ready(function() {
    $("p").eq(1);
});

// filter() method lets you specify a criteria. Elements that do not match
the criteria are removed from the selection, and those that match will be
returned.
$(document).ready(function() {
    $("p").filter(".intro");
});

// not() method returns all elements that do not match the criteria
$(document).ready(function() {
    $("p").not(".intro");
});
```

jQuery AJAX

What is AJAX?

AJAX = Asynchronous JavaScript and XML.

In short, AJAX is about loading data in the background and displaying it on the webpage, without reloading the whole page.

Examples of applications using AJAX: Gmail, Google Maps, Youtube, and Facebook tabs.

jQuery and AJAX?

jQuery provides several methods for AJAX functionality.

With the jQuery AJAX methods, you can request text, HTML, XML, or JSON from a remote server using both HTTP Get and HTTP Post - And you can load the external data directly into the selected HTML elements of your web page!

load()

```
// loads the content of the file "demo_test.txt" into a specific <div> element
$("#div1").load("demo_test.txt");
// loads the content of the element with id="p1", inside the file
"demo_test.txt", into a specific <div> element
$("#div1").load("demo_test.txt #p1");
```

load() callback

- `responseTxt` - contains the resulting content if the call succeeds
- `statusTxt` - contains the status of the call
- `xhr` - contains the XMLHttpRequest object

```
$("#button").click(function() {
    $("#div1").load("demo_test.txt", function(responseTxt,
statusTxt, xhr) {
        if(statusTxt == "success")
            alert("External content loaded successfully!");
        if(statusTxt == "error")
            alert("Error: " + xhr.status + ": " + xhr.statusText);
    });
});
```

Reference

https://www.w3schools.com/jquery/jquery_ref_ajax.asp

GET and POST

Two commonly used methods for a request-response between a client and server are: GET and POST.

- GET - Requests data from a specified resource
- POST - Submits data to be processed to a specified resource

GET is basically used for just getting (retrieving) some data from the server.
Note: The GET method may return cached data.

POST can also be used to get some data from the server. However, the POST method NEVER caches data, and is often used to send data along with the request. (https://www.w3schools.com/tags/ref_httpmethods.asp)

\$.get()

`$.get()` method requests data from the server with an HTTP GET request.

`$.get(URL, callback);`

```
$("button").click(function() {  
    $.get("demo_test.asp", function(data, status) {  
        alert("Data: " + data + "\nStatus: " + status);  
    });  
});
```

\$.post()

`$.post()` method requests data from the server using an HTTP POST request.

`$.post(URL, data, callback);`

```
$("button").click(function() {  
    $.post("demo_test_post.asp",  
    {  
        name: "Donald Duck",  
        city: "Duckburg"  
    },  
    function(data, status) {  
        alert("Data: " + data + "\nStatus: " + status);  
    });  
});
```