

jQuery Tutorial

What is jQuery?



- A framework for client-side JavaScript.
- Frameworks provide useful alternatives for common programming tasks.
- An open source project at jquery.com
- It simplifies
 - HTML document traversing
 - Event Handling
 - Animating
 - AJAX interactions

What is available with jQuery?

- Cross browser support and detection
- AJAX functions
- CSS functions
- DOM manipulation
- DOM transversal
- Attribute manipulation
- Event detection and handling
- JavaScript animation
- Hundreds of plugins for pre-built user interfaces, advanced animations, form validation, etc
- Expandable functionality using custom plugins
- Small foot print

Downloading jQuery

- Installation – You just download the latest jquery-x.y.z.js file and put it in your website folder
- <http://jquery.com/download>
- jQuery is lightweight: 90KB (minified and uncompressed), 33KB (Minified and Gzipped)
- Latest version is 3.5.1

So How Does jQuery Change How You Write JavaScript?

- jQuery adds a JavaScript object called **\$** or **jQuery** to your JavaScript code.
 - Through manipulation of this JavaScript code, it abstracts away commonly used JavaScript objects into **\$** and **jQuery**, such as the DOM (document), XMLHttpRequest, and JSON
- Example: Instead of
`var myButton = document.getElementById("myButton");`
- In jQuery, it's just
`$("#myButton");`

jQuery Basic Selectors

- These are examples of “Basic” selectors, based on CSS1:
- All Selector (“*”): selects all elements, sets css properties and returns the number of elements found

```
var elementCount = $("*").css("border", "3px solid red" ).length;
```

- Class Selector (“.class”): selects all elements with a given class and sets css properties

```
$(".myClass").css("border", "3px solid red");
```

- Element selector (“element”): selects all elements with the given tag name, e.g. div, and sets css properties

```
$("div").css("border", "9px solid red");
```

- ID selector (“#id”): selects a single element with the given id attribute

```
$("#myDiv").css("border", "3px solid red");
```

- Multiple selector (“selector1, selector2, selectorN”): selects a combined result of all the specified selectors

```
$("div, span, p.myClass").css("border", "3px solid red");
```

- For more examples see: <http://api.jquery.com/category/selectors/basic-css-selectors/>

Other jQuery Selector Categories

- JQuery borrows notation from CSS1-3 “selectors”, as a tool to match a set of elements. Here are some examples of what one can do:
- Attribute: selects elements that have the specified attribute and changes the associated text
`$("#input[value='Hot Fuzz'] "). text("Hot Fuzz") ;`
- Basic Filter, e.g. selects all elements that are h1, h2, h3, etc and assigns css properties
`$(":header").css({ background: "#ccc", color: "blue" });`
- Child Filter, e.g. finds the first span in each div and underlines the text
`$("#div span:first-child").css("text-decoration", "underline")`
- Content Filter, e.g. finds all div containing “John” and underlines them
`$("#div:contains('John')").css("text-decoration", "underline");`
- Form, e.g. finds all buttons and adds the css class “marked” to their properties
`var input = $(":button").addClass("marked");`
- For more examples see: <http://api.jquery.com/category/selectors>

jQuery Functions

- Either attached to the jQuery object or chained off of a selector statement.

- E.g. Run a function when the page is fully loaded

```
$( window ).load(function() {  
    //run code  
} );
```

- Most functions return the jQuery object they were originally passed, so you can perform many actions in a single line.

- E.g. Add the class *bigImg* to all images with height > 100 once the image is loaded

```
$("img.userIcon" ).load(function() {  
    if ( $( this ).height() > 100 {  
        $( this ).addClass("bigImg");  
    }  
});
```

- The same function can perform an entirely different action based on the number and type of parameters.

More jQuery Examples

- Remember these examples?

<http://csci571.com/examples.html#dom>

DOM Examples

- Example 1 → `document.getElementById.style.color`
- Example 2 → `document.getElementsByTagName`
- Example 3 → `document.getElementById().innerHTML`
- Example 4 → Moving Objects Horizontally
- Example 5 → Reversing Nodes in a Document
- Example 6 → DOM and Three innerhtml Examples
- Example 7 → DOM setting CSS Background Property
- Example 8 → DOM setting CSS Background Image Property
- Example 9 → DOM used for switching stylesheets

- We'll revisit the examples, but with jQuery instead!

Example 1:

document.getElementById.style.color

John slowly faded into view.

Fade Text

JavaScript w/o jQuery

```
hex=255 // Initial color value.
function fadetxt() {
  if(hex>0) { //If color is not black yet
    hex -= 11; // increase color darkness

    document.getElementById("sample").style.color="rgb("+hex+", "+hex+", "+hex+")";
    setTimeout("fadetxt()", 20);  }
  else    hex=255 //reset hex value
}
```

<http://csci571.com/examples/dom/ex1.html>

Example 1: \$.fadeOut(), \$.delay(), \$.fadeIn()

John slowly faded into view.

Fade Text

JavaScript with jQuery

```
$(function() { // when document is ready
    $("#fadeText").click(function() { // set a onClick handler on fadeText
        $("h3").fadeOut(125).delay().fadeIn(125);
        // fadeOut the h3 for 125 ms, delay, then fadeIn
    });
});
```

<http://csci571.com/examples/jquery/dom/ex1.html>

Example 2: document.getElementsByTagName

Font1

Font2

Font3

Font4

Count Font Tags

JavaScript w/o jQuery

```
function handleAllTags()
{
  var arrayOfDocFonts;
  if (document.all || document.getElementById) {
    arrayOfDocFonts = document.getElementsByTagName("font");
    alert("Number of font tags in this document are " + arrayOfDocFonts.length + ".");
  }
  else
    document.write("Unrecognized Browser Detected");
}
```

JavaScript w/ jQuery

```
$(function() { // when document is ready
  $("#countTags").click(function() { // when countTags is clicked,
    alert("Number of font tags in this document are " + $("font").length);
    // alert the number of font tags in the HTML
  });
});
```

<http://csci571.com/examples/dom/ex2.html>

<http://csci571.com/examples/jquery/dom/ex2.html>

Example 3:

document.getElementById().innerHTML

Number of clicks = 0

Increment Counter

JavaScript w/o jQuery

```
var hits = 0;
function updateMessage() {
    hits += 1;
    document.getElementById("counter").innerHTML = "Number of clicks = " + hits; }
```

JavaScript w/ jQuery

```
$(function() {
    var hits = 0;
    $("#updateMessage").click(function() {
        $("#counter").html("Number of clicks = " + ++hits);
    });
});
```

<http://csci571.com/examples/dom/ex3.html>

<http://csci571.com/examples/jquery/dom/ex3.html>

Example 4:

document.getElementById().style.left

Move Button right once

Move Button down Once

JavaScript and HTML w/o jQuery

```
<FORM>
<INPUT ID="counter1" STYLE="position:relative; left:0px" TYPE="button" VALUE="Move Button
right once"
    onclick="document.getElementById('counter1').style.left = '500px';">
</FORM>
<br><br><br><br><br><br><br><br>
<FORM>
<INPUT ID="counter2" STYLE="position:relative; top:0px" TYPE="button" VALUE="Move Button down
Once"
    onclick="document.getElementById('counter2').style.top = '15px';">
</FORM>
```

<http://csci571.com/examples/dom/ex4.html>

Example 4: \$.css();

Move Button right once

Move Button down Once

JavaScript and HTML w/ jQuery

```
<FORM>
<INPUT ID="counter1" STYLE="position:relative; left:0px" TYPE="button" VALUE="Move Button
right once"
    onclick="$('#counter1').css('left', '500px');">
</FORM>
<br><br><br><br><br><br><br><br>
<FORM><INPUT ID="counter2" STYLE="position:relative; top:0px" TYPE="button" VALUE="Move
Button down Once"
    onclick="$('#counter2').css('top', '15px');">
</FORM>
```

<http://csci571.com/examples/jquery/dom/ex4.html>

Example 5: document.getElementById(), parseInt()

Move Button

JavaScript w/o jQuery

```
var obj = document.getElementById('counter1');
var xlocation = parseInt(obj.style.left);
function handleClick( ) {
    xlocation += 50;
    document.getElementById('counter1').style.left = xlocation + "px";}
```

JavaScript w/ jQuery

```
$(function() {
    $("#counter1").click(function() {
        $("#counter1").css("left", (parseInt($("#counter1").css("left"))+50)+"px");
    });
});
```

<http://csci571.com/examples/dom/ex5.html>

<http://csci571.com/examples/jquery/dom/ex5.html>

Example 6: Uses childNodes, removechild, appendChild

paragraph #1

paragraph #2

paragraph #3

Click Me to Reverse

JavaScript w/o jQuery

```
function reverse(n)
{ // Reverse the order of the children of Node n
  var kids = n.childNodes; // Get the list of children
  var numkids = kids.length; // Figure out how many children there are
  for(var i = numkids-1; i >= 0; i--) { // Loop backward through the children
    var c = n.removeChild(kids[i]); // Remove a child
    n.appendChild(c); // Put it back at its new position
  } }
```

<http://csci571.com/examples/dom/ex6.html>

Example 6: \$.children(), \$.remove(), \$.append();

paragraph #1

paragraph #2

paragraph #3

Click Me to Reverse

JavaScript w/ jQuery

```
var onReady = function() {  
    $(".reverse").on("click", function() {  
        var kids = $("body").children();  
        for(var i = kids.length - 1; i >= 0; i--) {  
            var c = $(kids[i]).remove();  
            $("body").append(c);  
        }  
        onReady();  
    });  
}  
  
$(onReady);
```

<http://csci571.com/examples/jquery/dom/ex6.html>

Example 7: Uses innerHTML

HTMLInputElement:innerHTML

value:
set to:

Paragraph

Form

Division

JavaScript w/o jQuery

```
function setInnerHTML(nm, value) {  
  if (nm == '') return;  
  var  
element=document.getElementById?document.getElementById(nm):(document.all?document.all(nm):null);  
  if (element) {  
    if(element.innerHTML) {  
      element.innerHTML=value;  
    }  
    else notSupported( );  
  }  
  else NotSupported( );  
}
```

<http://csci571.com/examples/dom/domtest.html>

Example 7: \$.change() and \$.html();

HTMLInputElement:innerHTML

value:
set to:

Paragraph

Form

Division

JavaScript w/ jQuery

```
$(function() {  
    $("#sel").change(function() {  
        var selector = "#" + $("#sel").val();  
        $(selector).html($("#input[name='t']").val());  
    });  
});
```

<http://csci571.com/examples/jquery/dom/domtest.html>

jQuery & AJAX

- jQuery has a series of functions which provide a common interface for AJAX, no matter what browser you are using.
- Most of the upper level AJAX functions have a common layout:
 - **`$.func(url[,params][,callback])`**, [] optional
 - url: string representing server target
 - params: names and values to send to server
 - callback: function executed on successful communication.

jQuery AJAX load method

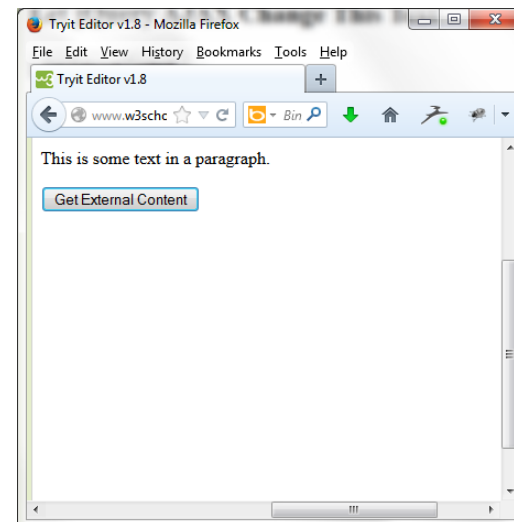
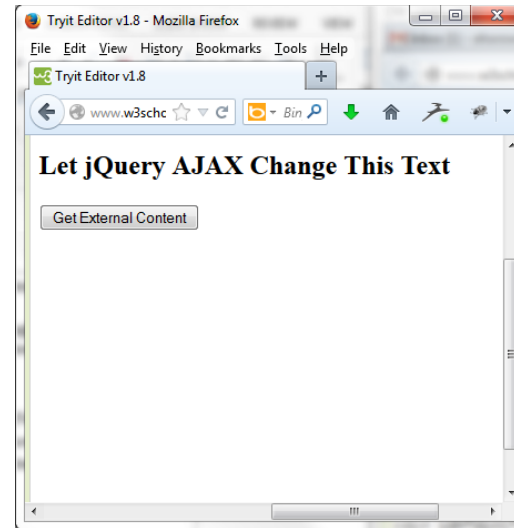
- The **load()** method loads data from a server and puts the returned data into the selected element.
- `$(selector).load(URL,[data,callback]);`
- The selector is usually a reference to div or span tag
- The required *URL* parameter specifies the URL you wish to load.
- The optional *data* parameter specifies a set of querystring key/value pairs to send along with the request.
- The optional *callback* parameter is the name of a function to be executed after the load() method is completed.
- For examples see

<http://csci571.com/ajaxexamples/simple/simpleajaxjquery.html>

or <http://csci571.com/examples.html#jquery> [4 examples under Ajax]

AJAX Example 1

```
<!DOCTYPE html><html><head>
<script
src="https://code.jquery.com/jquery-
3.4.1.min.js">
</script>
<script>
$(document).ready(function(){
    $("button").click(function(){
        $("#div1").load("demo_test.txt
#p1");
    });
});
</script>
</head><body>
<div id="div1"><h2>Let jQuery AJAX
Change This Text</h2></div>
<button>Get External Content</button>
</body></html>
```



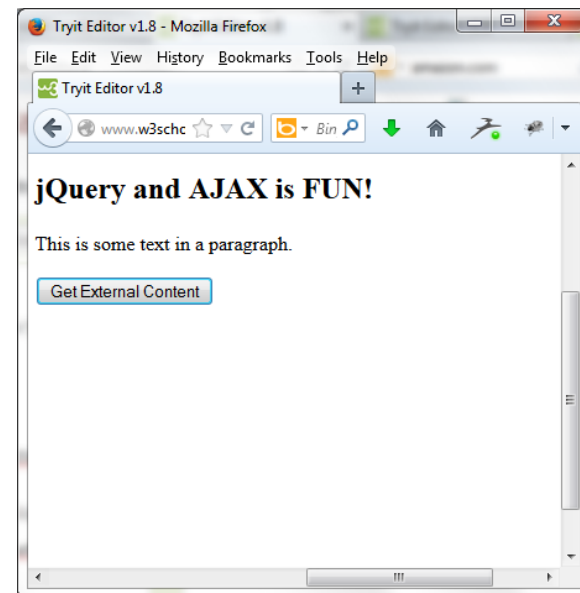
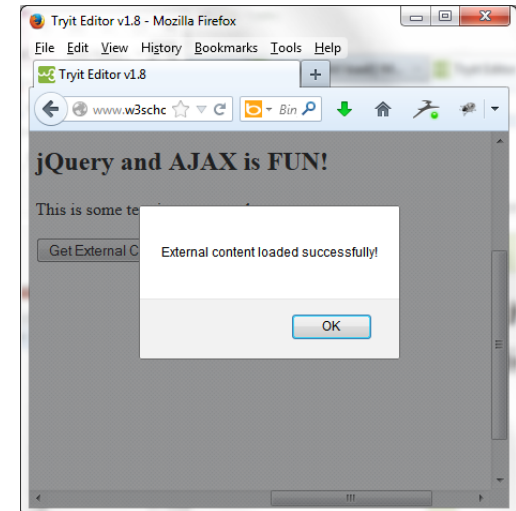
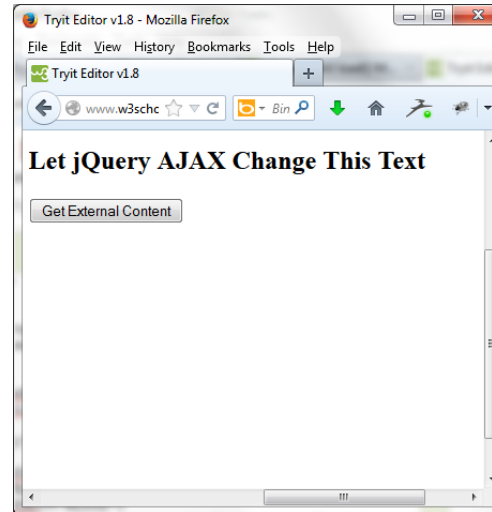
AJAX Example 2

```
<!DOCTYPE html><html><head>
<script
src="https://code.jquery.com/jquery-
3.4.1.min.js">
</script><script>
$(document).ready(function(){
    $("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);    });    });    });
</script></head><body>
<div id="div1"><h2>Let jQuery AJAX
Change This Text</h2></div>
<button>Get External
Content</button></body></html>
```

Note: non-jQuery version of .ready:

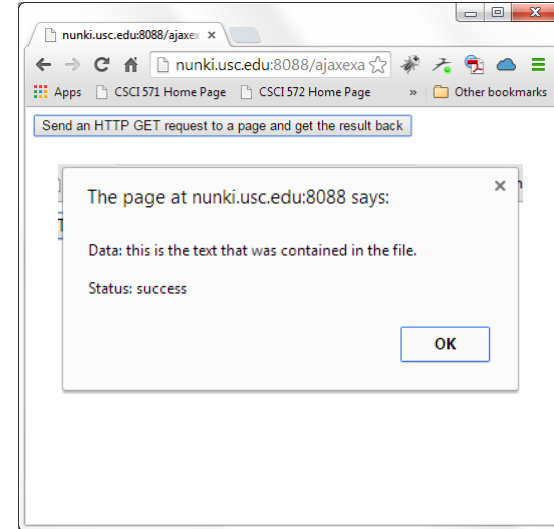
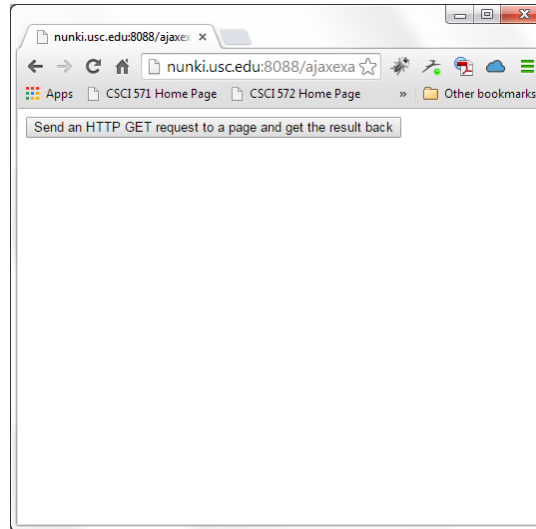
```
document.addEventListener("DOMContentLoaded",
function(event) { //do work });
```



AJAX Example 3 – GET Method

```
<!DOCTYPE html><html><head>
<script
src="https://code.jquery.com/jquery-
3.4.1.min.js">
</script><script>
$(document).ready(function(){
    $("button").click(function(){

$.get("demo_test_get.pl",function(data,
status){
    alert("Data: " + data +
"\nStatus: " + status);
    });
});
});
</script></head><body>
<button>Send an HTTP GET request to a
page and get the result back</button>
</body></html>
```



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

The required URL parameter specifies the URL you wish to request.

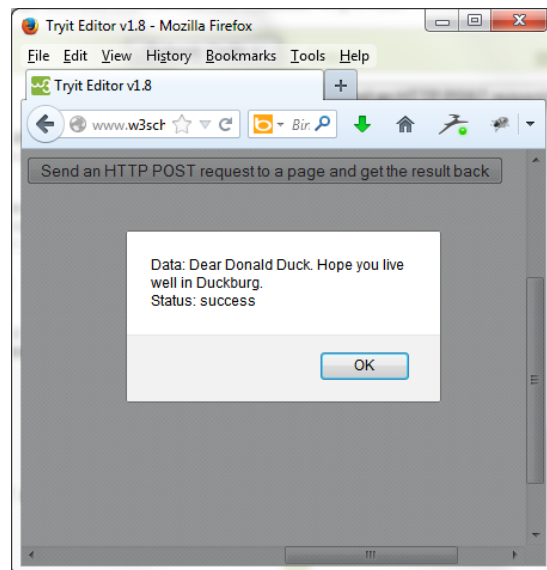
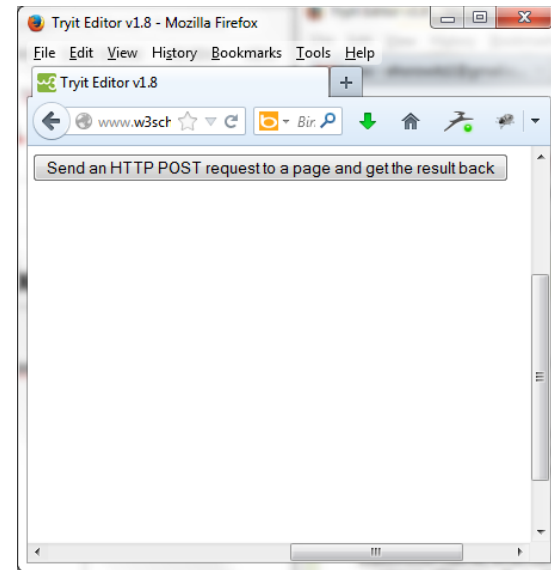
The optional callback parameter is the name of a function to be executed if the request succeeds.

The following example uses the \$.get() method to retrieve data from a file on the server

AJAX Example 4 – POST Method

```
<!DOCTYPE html><html><head>
<script src="https://code.jquery.com/jquery-
3.4.1.min.js">
</script><script>
$(document).ready(function(){
    $("button").click(function(){
        $.post("demo_test_post.php",
        {
            name:"Donald Duck",
            city:"Duckburg"
        },
        function(data,status){
            alert("Data: " + data + "\nStatus: " + status);
        });
    });
});
</script></head><body>
<button>Send an HTTP POST request to a page and get
the result back</button>
</body></html>

<?php
echo "Dear ".$_POST["name"]." Hope you live well in
".$_POST["city"];
?>
```



Summary jQuery AJAX Functions

- **\$.func(url[,params][,callback])**
 - \$.get
 - \$.getJSON
 - \$.getIfModified
 - \$.getScript
 - \$.post
- **\$(selector), inject HTML**
 - load
 - loadIfModified
- **\$(selector), ajaxSetup alts**
 - ajaxComplete
 - ajaxError
 - ajaxSend
 - ajaxStart
 - ajaxStop
 - ajaxSuccess
- **\$.ajax, \$.ajaxSetup**
 - async
 - beforeSend
 - complete
 - contentType
 - data
 - dataType
 - error
 - global
 - ifModified
 - processData
 - success
 - timeout
 - type
 - url

jQuery Usage Example (1) - Event

- jQuery way of a mouseover event that shows a submenu when menu is selected:

```
$( '#menu' ).mouseover(function() { // Anonymous function  
    $( '#submenu' ).show();  
});
```

jQuery Usage Example (2) - Event

- Stopping a normal **event** action: Suppose we want to stop the action of following a URL when a link is clicked. The action is part of the event object. We can reference the event object and call `.preventDefault()`;

```
$( '#menu' ).click(function(evt){  
    //JS code here  
    evt.preventDefault();  
})
```

jQuery Usage Example (3) – Form Selectors

- Selecting all form elements of a certain type:
\$(':text') It selects all text fields.
- Use with :input (all form elements), :password, :radio, :checkbox, :submit, :image, :reset, :button, :file, :hidden

See <https://api.jquery.com/category/selectors/form-selectors/>

- Set the value of a form element
Var fieldvalue = \$('#total').val(Yourvalue);
See <https://api.jquery.com/val/>

jQuery Usage Example (4) - Attribute

- Determining if checkbox is checked

```
If ($('#total').attr('checked')) {  
    //Do whatever you want if box is checked  
}  
else {  
    //Do whatever you want if box is not checked  
}
```

jQuery Usage Example (5) – Form Events

- Form Events such as submit:

```
$(document).ready(function() {  
    $('#signup').submit(function() {  
        if ($('#username').val() == '' ) {  
            alert ( 'Please supply name to name  
field' );  
            return false;  
        }  
    })  
});
```


jQuery Usage Example (6) - More events

- Focus Example: Auto erases default text in a field when it gets the focus

```
<input name="username" type="text" id="username"
value="Please type your user name">
$( '#username' ).focus(function() {
    var field = $(this);
    if(field.val()==field.attr( 'defaultValue' )) {
        field.val( '' );
    }
});
```

jQuery Usage Example (7)

- Click: If any radio button is clicked

```
$( ':radio' ).click(function() {  
    //do stuff  
});
```

- Add focus to the first element of the form:

```
$( 'username' ).focus;
```

jQuery Snippet that Detects Browser Width

```
<html><head><link rel="stylesheet" href="style.css" type="text/css" />
<script type="text/javascript" src="http://ajax.googleapis.com/ajax/libs/jquery/1.4.4/jquery.min.js">
</script></head><body>
<h2>Text that will be red if screen width is less than 800px, mobile.css is used,
and text will be green if screen width is greater than 800px, style.css is used, and black if no
style sheet is specified</h2>
<div id="browserInfo" style="padding:8px; border:1px solid blue; width:300px;"></div>
<script type="text/javascript">
    $(document).ready(function() {    resizeWindow();
        $(window).bind('resize', resizeWindow);
        getBrowserInfo();    });
function resizeWindow(){ $('#browserInfo').text('Browser (Width : ' + $(window).width() +
    ' , Height :' + $(window).height() + ' )');
        var newWindowWidth = $(window).width();
        // If window width is below 800px, switch to the mobile stylesheet
    if(newWindowWidth < 800){ $("link[rel=stylesheet]").attr({href : "mobile.css"});    }
        // Else if width is above 800px, switch to the large stylesheet
        else //if(newWindowWidth >= 800)
        {    $("link[rel=stylesheet]").attr({href : "style.css"});
            }
    }
function getBrowserInfo(){
        $('#browserInfo').text('Browser (Width : ' + $(window).width() +
    ' , Height :' + $(window).height() + ' )');    }
</script><noscript></body></html>
```

Is jQuery Worth It?

Yes	No
<p>Good use of the jQuery library will make it worthwhile in your code; will make JavaScript more readable and understandable</p>	<p>Bad use of jQuery library adds extra overhead. Why even add jQuery? Remember you need to add:</p> <pre data-bbox="884 611 1874 775"><script src="//ajax.googleapis.com/ajax/libs/ jquery/1.8.3/jquery.min.js"></script></pre>
<p>If web application requires a lot of DOM manipulation, hiding elements, fading out elements, etc</p>	<p>Doesn't even need DOM manipulation; could be done with CSS</p>
<p>Cross Browser Support – no need extra code for browser compatibility</p>	<p>Audience only uses Firefox – no need cross browser support only</p>

jQuery

- It's a useful library **when used wisely.**
- It will allow you to write JavaScript differently
 - **Write less, do more.**
- Remember: jQuery is just JavaScript
 - What you can do with jQuery, **you can always do without jQuery** but with *more code.*

jQuery Resources

- Project website
 - <http://www.jquery.com>
- Learning Center
 - <http://docs.jquery.com/Tutorials>
 - <http://www.learningjquery.com/>
 - <http://15daysofjquery.com/>
- Support
 - <http://docs.jquery.com/Discussion>
 - <http://www.nabble.com/JQuery-f15494.html> mailing list archive
 - irc.freenode.net irc room: #jquery
- Documentation
 - http://docs.jquery.com/Main_Page
 - <http://www.visualjquery.com>
 - <http://jquery.bassistance.de/api-browser/>
- jQuery Success Stories
 - http://docs.jquery.com/Sites_Using_jQuery