

Jquery notes:

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

- Embed into html with <script> tags
- Start with \$(document).ready(function(){
- To do something on click
 - \$("p").click(function(){
 - In quotes, put p for all <p> tags
 - Put #test for an element with id of test
 - Put .class for all elements in that class
 - (this) without quotes does to current element
- Somehow link to jquery in the header - options are:
 - <head>
 - <script src="jquery-3.7.1.min.js"></script>
 - </head>
 - src="<https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js>">
- Link path to jquery in the header
- Basic syntax is: \$(selector).action()
 - A \$ sign to define/access jQuery
 - A (selector) to "query (or find)" HTML elements
 - A jQuery action() to be performed on the element(s)
- Can embed into html or link to a file
 - <script src="my_jquery_functions.js"></script>
- Getting content
 - 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
 - Use \$(selector).text()
 - Getting value of an attribute
 - \$(selector).attr("href")
- Setting content
 - .text("text")
 - .attr("href", "hello") - sets attribute href to hello
- Changing elements
 - append() - Inserts content at the end of the selected elements
 - prepend() - Inserts content at the beginning of the selected elements
 - after() - Inserts content after the selected elements
 - before() - Inserts content before the selected elements
 - Unlimited values with commas separating
 - remove()

- empty()
- With css
 - addClass() - Adds one or more classes to the selected elements
 - removeClass() - Removes one or more classes from the selected elements
 - css(propertyname) - returns the property value of the first matched attribute
 - css(propertyname, propertyvalue) - sets that property to that value

AJAX

- load()
- Gets data from server and puts in the element
- Needs url/ path to file
- Example
 - `$("#div1").load("testingfile.txt")`
- Add callback parameter is optional
 - responseTxt - contains the resulting content if the call succeeds
 - statusTxt - contains the status of the call
 - xhr - contains the XMLHttpRequest object
- Example:


```

$("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);
    });
});

```
- Get
 - `$.get(URL, callback);`
 - 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 first callback parameter holds the content of the page requested, and the second callback parameter holds the status of the request.
 - Example:


```

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

```
- Post
 - `$.post(URL, data, callback);`
 - The required URL parameter specifies the URL you wish to request.

- The optional data parameter specifies some data to send along with the request.
- The optional callback parameter is the name of a function to be executed if the request succeeds.

- Example:

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

- a