**jQuery**

The $() is how we select an HTML tag. If we were to select the tag, we would have to write $('body'). This selector can grab any HTML tag, we will go into the different ways we can grab HTML items.

$('body').click(function(){

// more code will go here

});

$(document).ready(function(){

});

Inside the curly brackets of the function above is where you are to insert **all**of the jQuery functions. This code tells the browser to execute the jQuery functions when the document itself is ready. By *'ready'*, we mean when the document is fully loaded. ***This is very important because if you don't use this ready() function, your code will run before the HTML content you wrote gets rendered--meaning, the browser will run code for HTML that doesn't exist yet.***

**Select the HTML element/class/id:**

So in order to select all of the buttons of your web page, you type:

$('button')

That's it! Now you can add all sorts of cool functional properties to your buttons. What if you want to select all buttons with class name *blue*? Same as CSS!:

$('button.blue')

Easy! Now, if you want to select all buttons AND elements with class *blue*, do the following:

$('button, .blue')

And if you want to select an HTML with an ID, here is how we do that.

$('#red')

There you have the jQuery selector!

**Add an event handler**

Now that we have selected something using the jQuery selector, we can add an event listener. Remember, jQuery and JavaScript make a website interactive, and this is how we define what happens to those interactions. An **Event listener**is a trigger for all the code we wish to write that involves the element we selected. The event listener for a button being clicked:

$('button').click(); //\*\*now the document is listening on a click event for the button element!\*\*

Now, we have made jQuery listen for any button to get clicked. We have set an event listener for all buttons using jQuery.

**Write the action!**

The last portion is the action we want our page to enact when this event is triggered. So what happens now? What happens when we click a button on our page? Nothing yet because we haven't coded what we want to happen! Well, let's make something happen, let's add an action.

$('button').click(function(){

alert("You have clicked a button!");

});

Now, when you click a button on your webpage, you should see a pop up with the message we inserted. We put the **function(){}** code in to tell jQuery that we want to run a function when the button is clicked. Within that function, we write the code that makes the website interactive! If we want to pass parameters to the function, we specify them in the parenthesis. That's it! This is how you begin to incorporate jQuery into your websites. Take a few minutes to review the syntax for this function, as there are lots of parenthesis and brackets to account for.

### **Study only these tags!**

* **Effects**(functions to do some cool animation effects)
  + .hide()
  + .show()
  + .toggle()
  + .slideUp()
  + .slideDown()
  + .slideToggle()
  + .fadeOut()
  + .fadeIn()
* **CSS**(adding or removing a class for any HTML element/DOM)
  + .addClass()
  + .removeClass()
  + .css()
* **Manipulation**(retrieving or setting value or text in any HTML element)
  + .after()
  + .append()
  + .prepend()
  + .attr()
  + .before()
  + .html()
  + .text()
  + .val()
* **Events**(functions to handle an event)
  + .click()
  + .on()
  + .live() - deprecated (JQuery 1.7)
  + .hover()

Note that for the **manipulation** jQuery functions, there are few distinctions to make. The functions **.html()** and **.text()** are different in a key way: **.html()**can be used to insert new HTML markup, meaning new HTML tags. **.text()**is used to get or set just the **text**value of an **HTML element**. For instance, you could use **.text()**to change the text of a paragraph, but if you want to put an ordered list inside of the paragraph, you need to use **.html()**to insert the appropriate tags into the paragraph.

Similarly, **.append()**and **.html()** do nearly the same thing; they both can alter the HTML content of the selected item. The function **.append**will **add**markup to the element in question, whereas **.html()** will **overwrite**the markup with whatever is run inside the parentheses. So keep in mind that if you want to **add**content, use **.append()**, but if you want to **replace**content, use **.html()**.