jQuery

What is jQuery?

- A pre-written library of javascript routines that allows for routine scripting
- Adds transformative, interactive elements easily with little coding needed
- Has the ability to:
  - Isolate and manipulate HTML objects or CSS
  - Edit, change, and add html or css
  - Delay and tie execution to sync with events

Why jQuery?

- Increased compatibility
- Simple/streamlined

Basics of jQuery

```
<script src="http://code.jquery.com/jquery.js">
<script src="jquery.js"></script>
<script src="http://iyawebdev.com/jquery.js"></script>

//loading jQuery from the jQuery server or locally must be done before writing any code in
it
```

```
$(document).ready(function(){
  //code to be run
});

//used to start jQuery when the page loads; neccesary when using jQuery
```

- jQuery starts with a reference to an object (a selector), a period, then what you want to do

```
jQuery("header") //targets HTML header
jQuery("#box1") //targets HTML object with ID box1
$(".headline") //targets HTML object with class headline
$(this) //targets the object under reference

//the various ways of targeting items in jQuery
```

```
$("#secondp").hide();

//targets #secondp and hides it
```

Common jQuery Actions

```
.addClass("myClass")
//add a CSS class to an object
```

```
.append()
//add content to the end of an element

<div class='myDiv'>
  <img tag thats already here>
  //this is where it will go
</div>
//in a script tag
<script>
  $(".myDiv").append("<img src='pic.png'>")
</script>
```

```
.attr()
//retrieve or set the value of an HTML object

.attr("src", "texttextext") //will sent font color to green

.attr("src") //will tell you CSS color property
```

```
.css()
//retrieve or set the value of an a CSS property

//with one parameter, it reads the property
//two parameters it will replace

.css("color", "green") //will sent font color to green

.css("color") //will tell you CSS color property
```

```
.html()
//sets the html content of a page

.html("content")
```

```
.fadeIn()
//fade in an object

//takes time in ms as an argument
```

```
.fadeOut()
//Fade out an object
```

```
.fadeToggle()
//Alternate fading in or out
```

```
.hide()
//make an element disappear
```

```
.next()
//from that object, select the next HTML object
```

```
.prev()
//from that object, select the previous
```

```
.prepend()
//add content to the front/beginning of an element
```

```
.removeClass()
//remove a CSS class from an object
```

```
.show()
//make an element appear
```

```
.slideDown()
//Reveal a hidden object
```

```
.slideUp()
//animate out an object
```

```
.slideToggle()
//Alternate revealing and hiding an item
```

```
.toggle()
//alternate hiding and showing an element
```

```
.toggleClass()
//add or remove a class (alternating) from an object
```

```
.delay()
//allows scripted pauses
```

```
.val()
```

```
$("#textbox").val()
$("#textbox").val("replace the value")
```

jQuery Manipulations

```
$("").on("")
//.on triggers when a certain selector has something done to it
//creates a function of code to be executed later
//can be when a certain object with class "class1" has a mouse hover over it
```

```
$(".secondp").on("hover", function(){
  //code the executed when the user hovers
});

document.querySelector(".secondp").addEventListener("mouseover", fucntion(){

}
//creates a block that triggers when the user hovers
```

```
$("#secondp").on("mouseout",function() {
      $("#secondp").removeClass("greenonyellow");
  })
//when hovering, it's important to realize that there has to be a mouseout function when applying something, since, once the hover occurs, its been applied

//mouseout triggers when the mouse leaves an object
```

```
$("#secondp").on("mouseover", function(){
  //code the executed when the user moves their mouse over an object
});

//mouseover triggers when the mouse moves over an object
```

```
$("li").on("mouseover",function() {
      $(this).addClass("large");
  })

//when doing a statement like this, it explains that whenever you hover over an li, it should trigger that code; however, targeting "li" will effect every li on the page
```

```
//the keyword "this" only targets what was hovered only and only effects that
```

```
$("img").on("mouseover", function(){
    $(this).next().toggleClass("larger");
  });

$("img").on("mouseover", function(){
    $(this).prev().toggleClass("larger");
  });
//.next() targets the next HTML object in the code; so it doesnt trigger on the image you are over, but the next object in the HTML

//.prev() targets the previous

//this makes use of chaining, in which multiple elements are chained together in a row and are executed in order
```

```
Events to tbe used in jQuery with on commands
"mouseover" //when mouse hovers over
"mouseout" //when mouse hovers out
"click" //when mouse clicks
"dblclick" //when mouse double clicks
"blur"
"focus"
"change"
```