WMDD 4835 Summer 2017 Assignment 6 Part 1

Topics:

- **jQuery**
  - Creating new HTML Elements
  - .closest()
  - .attr()
  - .append()
  - .prepend()
  - .remove()
  - .siblings()
  - .find()
  - .on("click")
  - $(this)
  - event.preventDefault()

Optional Topics:

- **jQuery**
  - .offset()
  - .scrollTop()
  - .on("scroll")
  - .on("load")

Before trying this assignment, you may want to complete the exercises on the jQuery tutorial page: try.jquery.com

Download the starter files from D2L. I've given you the HTML for a basic **random image gallery** and **random image generator**, using the random place-holder image capabilities of Unsplash.com. **Do not modify the HTML except where directed to.**

Your job is to use jQuery to implement the functionality of clicking the "generate random image" button to create a new image, and "Add to Gallery" to add this image to the gallery of images. There is a random image generator widget at the top and the bottom of the page.

Instructions:

- I'm not too worried about the styling for this assignment (as you can see from the demo video), but use **flexbox** to lay out the image gallery. This is a similar layout to Practice Assignment B. Use **object-fit** and **flex-grow** to get the images to horizontally fill the container.
- Use a **click event handler** to detect when the user clicks on the "Generate Random Image" element.
  - Prevent the default behavior of clicking on an anchor tag by passing the **event** variable into the handler function and using **event.preventDefault().**

    - Use the **.remove()** method to remove the current image in the generator. To target the **img** element, use traversal methods like **.siblings(),** .**closest()**, or **.find(). Do not use CSS selectors.**
    - Create a new image tag. Consult this resource for information about how to do this: https://www.w3schools.com/jquery/jquery_dom_add.asp
    - Add a **src** attribute to the **img** tag, using the **Unsplash** random place-holder generator. Instructions for the format of the URL can be found here: https://unsplash.it/
        - Use a height of 250px and a random width that falls within some reasonable range. You will need to use JavaScript **Math methods** and **string operators** to do this.
    - Use the **.append()** method to add this new image to the **.image-generator div**. Do this **without using the class selector**. You may need to use the **.closest()** method to get the closest **parent div** of the button that triggered the event.
- Implement the functionality for the "Add to Gallery" button.
    - Create a jQuery object for the **.image-area** using its class selector (or some other method if you want to challenge yourself) and use the **.prepend()** method to add the image to the gallery.
    - Create another random image to fill the empty space in the image generator area, using the above methods.

Optional Challenges:

- Try implementing a "loading" screen with a spinning icon that displays while the Unplash images are loading.
    - **You may add your own HTML for this part.**
    - To do this, you can use the **$(window).on("load", function(){})** method to detect when the page has loaded with all images. This is different than using **$(document).ready()** because it actually detects when images have finished loading.
    - To create the animation, you can use jQuery to add classes to the elements, and use CSS transitions.

- Try implementing a visual effect that causes each image in the gallery to fade in and float up as the user scrolls down the page. See the demo movie for an example of this.
    - To do this, you can use the **$(window).on("scroll", function(){})** method to detect scroll events, and get the scroll position in the document by using **$(window).scrollTop()**. You can get the coordinates of elements within the document by using **$("selector").offset().** You may need to use **$("selector").each(function(){})** to execute a function on each matched element in a jQuery object.
    - To create the animation, you can use jQuery to add classes to the elements, and use CSS transitions.