

## Lab 07: Working with jQuery

### Objective(s):

1. Learn Traversing using jQuery
2. Learn Effects in jQuery
3. Learn jQuery Events
4. Learn Event Delegation using jQuery
5. Learn about Bubbling
6. Learn preventDefault & stopPropagation

### Lab Task(s):

#### Exercises

1. Complete the following exercises for the following HTML page. Make sure to do this ENTIRELY using jQuery!

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
  <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.cs
s" integrity="sha384-
BVYiISIFeK1dGmJRAkycuHAHRg320mUcww7on3RYdg4Va+PmSTsz/K68vbdEjh4u"
crossorigin="anonymous">
  <style>
    img {
      max-width: 70%;
    }

    .image-center {
      display: block;
      margin: 10px auto;
    }
  </style>
</head>
<body>
  <div class="container">
    <h1 id="title" class="text-center">Welcome to my website.</h1>
```

```

<div class="row">
  <article class="col-sm-8">
    <p>My website is very important because I have many important
things to say. Here is a picture of a puppy in a cup: </p>
    
    <p>Here are some more thoughts: Lorem ipsum dolor sit amet,
consectetur adipisicing elit. Cum harum labore veritatis, possimus facilis
quas repellat, explicabo, ad commodi ipsum quidem nostrum earum pariatur
fugit! Magni sunt nulla impedit! Facilis.</p>
    <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. Eius,
natus qui nostrum libero atque hic distinctio sint facilis nihil possimus
minus tenetur deserunt rerum laboriosam cum quisquam doloremque ducimus
error.</p>
    <p>Rerum debitis aspernatur, ipsum animi! Quis ullam dolore
blanditiis accusantium eius minus tempore, iure maxime similique vel!
Possimus fugiat reprehenderit autem, recusandae quos laudantium nesciunt
libero suscipit officia rerum, et.</p>
    <p>Aliquam harum at soluta id esse perspiciatis, nesciunt corporis
nostrum commodi, dolorum, repellat possimus a cumque. Id sed explicabo,
quaerat sit esse facere, voluptatem, odit maxime nesciunt eveniet
assumenda, error!</p>
    <p>Consequuntur porro, nobis placeat, labore magni aliquam
repellendus autem consequatur hic facere, molestiae dolorum suscipit quos
sint, minima ipsum. Quibusdam voluptatem est hic, corporis. Officiis
delectus, consectetur harum reiciendis perferendis.</p>
  </article>
  <aside class="col-sm-4">
    <h4>I love lists. Here's why.</h4>
    <ol>
      <li>Sometimes they're numbered, and numbers are great.</li>
      <li>Sometimes they're not, and that's okay too.</li>
      <li>I don't have the attention span to read paragraphs.</li>
      <li>Hey look, there's a puppy!</li>
    </ol>
  </aside>
</div>
<div class="row">
  <div class="col-sm-4">
    Red
    <input class="form-control" type="number" min="0" max="255"
step="1" value="255">
  </div>
  <div class="col-sm-4">
    Blue
    <input class="form-control" type="number" min="0" max="255"
step="1" value="255">
  </div>
  <div class="col-sm-4">
    Green
    <input class="form-control" type="number" min="0" max="255"
step="1" value="255">
  </div>
</div>

```

```

        </div>
    </div>
</div>
<script src="https://code.jquery.com/jquery-3.1.1.js" integrity="sha256-
16cdPddA6VdVInumRGo6IbivbERE8p7CQR3HzTBuELA="
crossorigin="anonymous"></script>
</body>
</html>

```

1. When the DOM is ready, console.log the message "Let's get ready to work with jQuery!"
  2. Give all images inside of an article tag the class of image-center (this class is defined inside of the style tag in the head).
  3. Remove the last paragraph in the article.
  4. Set the font size of h1 with an id of title to be a random pixel size from 0 to 100.
  5. Add an item to the list; it can say whatever you want.
  6. Scratch that; the list is silly. Empty the aside and put a paragraph in it apologizing for the list's existence.
  7. When you change the numbers in the three inputs on the bottom, the background color of the body should change to match whatever the three values in the inputs are.
  8. Add an event listener so that when you click on the image, it is removed from the DOM.
2. Given the following HTML.

```

<button id="clicker">Click me to show a hidden secret!</button>
<div id="secret" style="display: none">I am a hidden secret.</div>

```

- a. Add jQuery code that fades in the text "I am a hidden secret" whenever the button is clicked.
  - b. Change your jQuery code so it slides down the text instead of fading it in.
  - c. Change the button text to say "Click me to toggle a hidden secret!" and make the text toggle between fading in and out each time it is clicked.
3. Given the following HTML & JS.

### HTML

```

<span>Enter your name:</span>
<input type="text" id="name">
<button id="greet">Greet me!</button>

```

## JavaScript

```
// Call say("Hi!") to have your computer say hi!  
// This only works on recent versions of Safari  
// and Chrome at the moment.  
function say(text) {  
    var msg = new SpeechSynthesisUtterance(text);  
    window.speechSynthesis.speak(msg);  
}
```

1. Add jQuery that greets the user by calling the above say function whenever the user clicks the "Greet me!" button, taking into account the value of the text field.
2. For example, if the user types the word "Bob" into the text field, then the computer should say "Hello Bob!" when the button is clicked.
4. Given the following HTML.

```
<button id="clicker">Click me to hide the hidden secrets!</button>  
<p class="secret">I am hidden secret #1.</p>  
<p class="secret">I am hidden secret #2.</p>  
<p class="secret">I am hidden secret #2.</p>  
<p>I am <em>not</em> a hidden secret.</p>
```

Add jQuery code to modify the page so that when the user clicks the button, the paragraphs that start with the words "I am a hidden secret" slide up.

5. Given the following HTML.

```
  
<p>  
    <button id="clicker">Click me to change the picture</button>  
</p>
```

Change the page with jQuery so that when the button is clicked, the image changes to another.

You can store different images' url or your own choice in an array and render them randomly.

6. Build a list page where users can add and rate examples from a category (e.g. movies, albums, or sports). This page should include the following:
  - a. A form, where you can add something to the list and rate it.

- b. A table of all of the things you've added.
- c. A delete button for each row of the table that lets you remove elements from the list.
- d. A sorting feature, so you can sort entries in the table by their title or their rating. (To implement this you may need to research JavaScript's sort method.)
- e. Title should be required. If user doesn't enter a value, then on button (Add Movies) click text box's border should be red and focused. Otherwise should be normal.
- f. Here's an example of a movie list page (it includes Bootstrap for styling):

**My Favorite Movies**

Title  Rating

| Title         | Rating | Delete                                |
|---------------|--------|---------------------------------------|
| Gigli         | 4      | <input type="button" value="Delete"/> |
| Thor          | 6      | <input type="button" value="Delete"/> |
| Wayne's World | 8      | <input type="button" value="Delete"/> |
| The Matrix    | 9      | <input type="button" value="Delete"/> |

7. Apply jQuery on the following form for the validation which you developed earlier in the Bootstrap exercises.

First name: Mark (Looks good!)

Last name: Otto (Looks good!)

Username: @Username (Please choose a username.)

City: City (Please provide a valid city.)

State: State (Please provide a valid state.)

Zip: Zip (Please provide a valid zip.)

☐ Agree to terms and conditions  
You must agree before submitting.

[Submit form](#)

8. Build a simple page showing Ryu firing a Hadouken. Use jQuery to switch images when mouse hovers or clicks on Ryu and animate a Hadouken image on click. Display the total number of Hadoukens fired as text below the image. The goal is to code the mouse interactions using jQuery inside the index.html file as shown in the demo video.
  - a. When mouse enters the Ryu image, the **images/ryu\_animated.gif** image shall be shown.
  - b. When mouse leaves the Ryu image, the initial still image shall be shown.
  - c. When mouse button is pressed, the image **images/ryu\_hadouken\_pose.png** shall be shown plus:
    - i. A new image **images/hadouken.gif** shall be added to the DOM
    - ii. The hadouken image shall be animated to the right using a 'swing' animation of one second.
    - iii. When the animation is done, the hadouken image shall be removed from the DOM
  - d. The number of hadoukens fired by Ryu shall be counted and displayed as text below the image. Update the counter after the hadouken was been fired. For solving this exercise, look up the jQuery functions **hover**, **mousedown**, **mouseup** and **animate**.

New DOM elements are created with this syntax in jQuery:

```
var image = $('<img>', {src: 'https://...', alt: 'New Image'});
```

Images are uploaded on LMS.