	CSE 482L: Internet and Web Technology Lab	Lab Manual
	Faculty: naqib.hussain@northsouth.edu Instructor: shabbir.ahmed01@northsouth.edu Updated: 2023-Nov-13	5

Goal**:** Gain experience on JQuery[a]

- Sourcing and placement
- Syntax, selectors, hide/show, callback, chaining
- Get/set, add / remove elements, add/remove CSS classes
- Event handling, document ready function

JQuery Basics, Sourcing and Placement

Try the following file and test with (a) `<script src="https:….js">` placed at the bottom of the `<body>` and then (b) try it with the button click action place above the <script src=".."">.

File name: **sourcing_jquery.html** [b]

```html
<!DOCTYPE html>
<html>
<head>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js">
    </script>
</head>
<body>
    <h2>This is a heading</h2>
    <p>This is a paragraph.</p>
    <p>This is another paragraph.</p>
    <button>Toggle paragraphs</button>
    <script>
        $("button").click(function () {
            $("p").toggle();
        });
    </script>
</body>
</html>
```

Why and how it behaves for case (a) and case (b)? What works what does not? Why? Can you explain?

Syntax and Selectors

File name: **syntax_demo_jquery.html**

```html
<!DOCTYPE html>
<html>
<head>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js">
    </script>
</head>
<body>
    <h2 id="idheading">This is a &lth&gt. click here to hide it</h2>
    <p class="vanish">This is a &ltp&gt.its has .vanish class</p>
    <p>This is another &ltp&gt.</p>
    <p id="vanish">This is a &ltp&gt. its has .vanish id</p>
    <button class="vanish">Toggle &ltp&gt (has vanish class)</button>
    <button>Toggle paragraphs</button>
    <input id="idvanishbyclass" type="button" value="Toggle vanish class">
    <input id="idvanishbyid" type="button" value="Toggle vanish id">
    <script>
        $("button").click(function () {
            $("p").toggle();
        });
        $("#idheading").click(function(){
            $(this).hide();
        });
        $("#idvanishbyclass").click(function(){
            $(".vanish").toggle();
        });
        $("#idvanishbyid").click(function(){
            $("#vanish").toggle();
        });
    </script>
</body>
</html>
```

Interact with the buttons and inputs and paragraphs. Try to explain each ones response.

`$(document).ready()` Function

This function ensures that any jQuery code runs only after the HTML DOM ie the document loading is done and its ready. The following example functions exactly the same as above one. Although, you source your JQuery code from an external file in the head it will load prior to the body it will not access any DOM element until the DOM is loaded ready.

File name: **document_ready_demo.html**

```html
<!DOCTYPE html>
<html>
<head>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js">
    </script>
    <script src="myscript.js"></script>
</head>
<body>
    <h2 id="idheading" class="vanish">This is a &lth&gt. click here to hide it. it
also has .vanish class</h2>
    <p class="vanish">This is a &ltp&gt.its has .vanish class</p>
    <p>This is another &ltp&gt.</p>
    <p id="vanish">This is a &ltp&gt. its has .vanish id</p>
    <button class="vanish">Toggle &ltp&gt (has vanish class)</button>
    <button>Toggle paragraphs</button>
    <input id="idvanishbyclass" type="button" value="Toggle vanish class">
    <input id="idvanishbyid" type="button" value="Toggle vanish id">
</body>
</html>
```

File name: **myscript.js**

```javascript
$(document).ready(function () {
    $("button").click(function () {
        $("p").toggle();
    });
    $("#idheading").click(function () {
        $(this).hide();
    });
    $("#idvanishbyclass").click(function () {
        $(".vanish").toggle();
    });
    $("#idvanishbyid").click(function () {
        $("#vanish").toggle();
    });
});
```

Events Handling

```html
<!DOCTYPE html>
<html>
<head>
    <style>
        div {
            border: 3px solid black;
            height: 4rem; width: 20rem;padding-top:2rem;
            text-align: center;
        }
    </style>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js">
    </script>
</head>
<body>
    <div>Click or move the mouse pointer over this &ltdiv&gt</div>
    <script>
        $(document).ready(function () {
            $("div").on({
                mouseenter: function () {
                    $(this).css("background-color", "lightgray");
                },
                mouseleave: function () {
                    $(this).css("background-color", "lightblue");
                },
                click: function () {
                    $(this).css("background-color", "yellow");
                }
            });
        });
    </script>
</body>
</html>
```

Also notice how you can manipulate CSS style attributes of DOM elements

Callback

Pass a function as argument to another function.

File name: **`callback_demo.html`** ^d

```html
<!DOCTYPE html>
<html>
  <head>
    <style>
      p {
        border: solid;
        height: 2rem;width: 10rem;
        text-align: center;
      }
    </style>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js">
    </script>
    <script>
      $(document).ready(function () {
        $("button").click(function () {
          $("p").hide(1000, function () {
            alert("The <p> already hidden now");
          });
        });
      });
    </script>
  </head>
  <body>
    <p>&ltp&gt to be hidden.</p>
    <button>Hide</button>
  </body>
</html>
```

As expected, it hides the paragraph first and then shows alert. The `alert()` is called inside a callback function which is given as an argument to the `hide()` function. If you want to see what happens without a call back, then modify the above code as follows.

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      p {
        border: solid;
        height: 2rem;width: 10rem;
        text-align: center;
      }
    </style>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js">
    </script>
    <script>
      $(document).ready(function () {
        $("button").click(function () {
          $("p").hide(1000);
          alert("The <p> already hidden now");
        });
      });
    </script>
  </head>
  <body>
    <p>&ltp&gt to be hidden.</p>
    <button>Hide</button>
  </body>
</html>
```

This time the alert shows up before the `<p>` hides away.

Method Chaining

Method chaining is a performance booster. It allows execution of multiple methods on the same object without searching for the object multiple times across the DOM document. Specially the DOM is large (number of elements exceeding 1000) such chaining may produce significant performance improvement.

File name: **chaining_demo.html** ᵉ

```html
<!DOCTYPE html>
<html>
  <head>
    <style>
      p {
        border: solid; text-align: center;
        height: 2rem; width: 10rem;
      }
    </style>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js">
    </script>
    <script>
      $(document).ready(function () {
        $("button").click(function () {
          $("#p1").css("color", "red").slideUp(2000).slideDown(2000);
        });
      });
    </script>
  </head>
  <body>
    <p id="p1">faith, peace, prosperity</p>
    <button>slide</button>
  </body>
</html>
```

Try the same example without method chaining on the $("#p1"). Now for this short DOM the performance improvement may not be perceptible. You should consider adopting this while working in your own project. A good way to feel the performance improvement is apply chaining inside methods responding to scroll event.

Get / Set Elements

File name: **get_value_set_text_demo.html**

```html
<!DOCTYPE html>
<html>
  <style>
    div { border: solid black; text-align: center; height: 2rem; width: 20rem;}
  </style>
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js">
  </script>
  <body>
    <h2>JavaScript Regular Expressions</h2>
    <div id="notif"></div>
    <form action="#">
      <p>Input a valid email address or a BD mobile number</p>
      <label for="email_mobile"></label>
      <input type="text" id="email_mobile" /><br /><br />
      <button type="button" onclick="my_checkValidity();">Test validity</button>
    </form>
    <script>
      function my_checkValidity() {
        let mobile = /^\+?(88)?0?1[3456789][0-9]{8}\b/i;
        let email = /^[\w-\.]+@([\w-]+\.)+[\w-]{2,4}$/i;
        if ($("#email_mobile").val().match(mobile)) {
          $("#notif")
            .text("Valid mobile: " + $("#email_mobile").val().match(mobile))
            .css("backgroundColor", "lightgreen");
        } else if (text.match(email)) {
          $("#notif")
            .text("Valid email: " + $("#email_mobile").val().match(email))
            .css("backgroundColor", "lightgreen");
        } else {
          $("#notif").text("Not valid!").css("backgroundColor", "red");
        }
      }
    </script>
  </body>
</html>
```

Same methods `text(), html()` and `val()` can be used to get and set content. To learn more explore the link given at the end notes section. **f**

Add / Remove Elements

```html
<!DOCTYPE html>
<html>
  <head>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js">
    </script>
    <script>
      $(document).ready(function () {
        $("#btn1").click(function () {
          $("p").prepend("<b>Prepended text</b>. ");
        });
        $("#btn2").click(function () {
          $("ol").append("<li>Appended item</li>");
        });
        $("#btn3").click(function () {
          $("p").before("<b>Before</b>");
        });
        $("#btn4").click(function () {
          $("ol").after("<i>After</i><br>");
        });
        $("#btn5").click(function () {
          $("p").remove();
        });
        $("#btn6").click(function () {
          $("ol").empty();
        });
      });
    </script>
  </head>
  <body>
    <p>This is a paragraph.</p>
    <p>This is another paragraph.</p>
    <ol>
      <li>List item 1</li>
      <li>List item 2</li>
    </ol>
    <button id="btn1">Prepend in &ltp&gt</button>
    <button id="btn2">Append in &ltp&gt</button>
    <button id="btn3">Insert before &ltp&gt</button>
    <button id="btn4">Insert after &ltol&gt</button>
    <button id="btn5">Remove &ltp&gt</button>
    <button id="btn6">Empty &ltol&gt</button>
  </body>
</html>
```

append() - Inserts content at the end. prepend() - Inserts content at the beginning. after() - Inserts content after the selected elements. before() - Inserts content before the selected elements. remove() - Removes an element along with its child elements. empty() - Removes the child elements. To learn more visit the links at the end notes section.

Add / Remove CSS classes

File name: **get_value_set_text_demo.html**

```html
<!DOCTYPE html>
<html>
  <head>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js">
    </script>
    <style>
      .important {
        font-weight: bold;
        font-size: xx-large;
      }
      .blue {
        color: blue;
      }
    </style>
  </head>

  <body>
    <h1>Heading 1</h1>
    <h2>Heading 2</h2>
    <p>This is a paragraph.</p>
    <div>This is some important text!</div><br/>
    <button id="btn1">Add classes to elements</button>
    <button id="btn2">Add classes to elements</button>

    <script>
      $(document).ready(function () {
        $("#btn1").click(function () {
          $("h1, h2, p").addClass("blue");
          $("div").addClass("important");
        });
        $("#btn2").click(function () {
          $("h1, h2, p").removeClass("blue");
          $("div").removeClass("important");
        });
      });
    </script>
  </body>
</html>
```

For more follow link at the end note.[h]

Handling Cookies

There is not specific established JQuery function to handle cookies. Following codes demonstrate how you can handle cookies from the front-end ie the browser. Make sure to use the correct path to `eatcookies.js` file as sourced in the html.

File name: **cookie_handling_front_end_demo.html** [i]

```html
<!DOCTYPE html>
<html>
  <head>
    <style>
      #idnotif {
        border: solid black 2px;
        height: 2rem; width: 20rem;
        text-align: center;
      }
    </style>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script>
    <script src="eat_cookies.js"></script>
  </head>
  <body>
    <h2>This page sets and gets cookie</h2>
    <p id="idnotif"></p>
    <button>Clear cookie</button><br><br>
    <i>
      On DOM ready event this page will set a non HTTPOnly and non secured
      cookie. Open DevTool, goto Application tab. Choose Cookies menu on the
      left panel. You should be able to ready and delete cookies directly now.
    </i>
  </body>
</html>
```

This following JS file is based on an example from W3school with minor modification for convenience of demonstration.

File name: **eat_cookies.js**

```javascript
$(document).ready(function () {
  checkCookie();
  $("button").click(function () {
    setCookie("username", "", 12);
  });
});
// continue to the next page to find the rest…
```

```javascript
function setCookie(cname, cvalue, exdays) {
  const d = new Date();
  d.setTime(d.getTime() + exdays * 24 * 60 * 60 * 1000);
  let expires = "expires=" + d.toUTCString();
  document.cookie = cname + "=" + cvalue + ";" + expires + ";path=/";
  $("#idnotif").text("Cookie is set: " + cname + " = " + cvalue);
}

function getCookie(cname) {
  let name = cname + "=";
  let decodedCookie = decodeURIComponent(document.cookie);
  let ca = decodedCookie.split(";");
  for (let i = 0; i < ca.length; i++) {
    let c = ca[i];
    while (c.charAt(0) == " ") {
      c = c.substring(1);
    }
    if (c.indexOf(name) == 0) {
      return c.substring(name.length, c.length);
    }
  }
  return "";
}

function checkCookie() {
  let user = getCookie("username");
  if (user != "") {
    $("#idnotif").text("Welcome again: " + user);
  } else {
    user = prompt("Please enter your name:", "");
    if (user != "" && user != null) {
      setCookie("username", user, 30);
    }
  }
}
```

On DOM ready event this page will set a non HTTPOnly and non-secured cookie. Open DevTool, goto Application tab.
Choose Cookies menu on the left panel. You should be able to ready and delete cookies directly now. For better
navigation see Figure **1**.This snapshot may help you locate cookie.
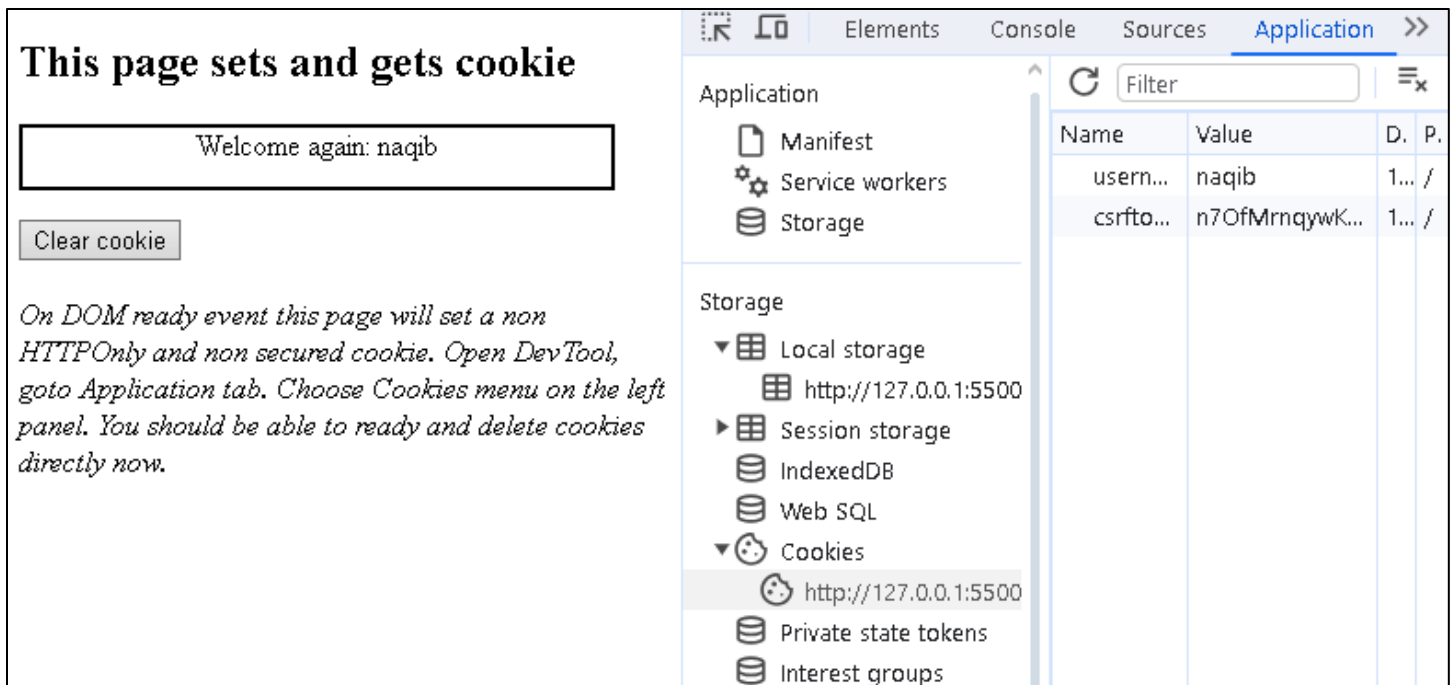
Figure 1: Use browser's DevTool to locate cookie.

Lab Task

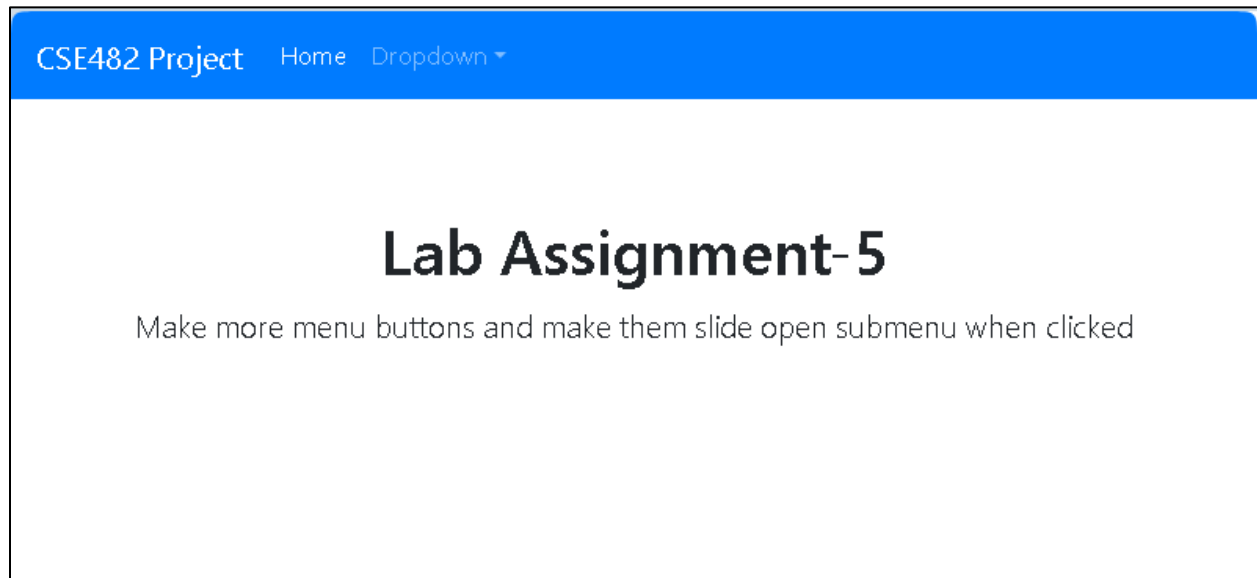(a) Take your form validation assignment done in Lab-04. Replace the plain JS code as by JQuery code.

(b) Add a "**remember me**" check box to the above form. When checked save, programming skills and college information in a cookie.

Assignment

Make a menu on a web page below and make it slide open when the menu button gets a click or a tap. Feel free to choose your own color scheme, use bootstrap or add more menu items. You may add a search form also. In addition, you may make it a part of your semester final project.

Practice to Learn More

Learn examples at W3schools (https://www.w3schools.com/jquery/default.asp) as much as possible. However, run examples from the following URLs and then make changes in the code, then bring them in the next class in a pen drive or google drive.

A. https://www.w3schools.com/jquery/jquery_dimensions.asp
B. https://www.w3schools.com/jquery/jquery_traversing.asp
C. https://www.w3schools.com/jquery/jquery_ajax_intro.asp
D. https://www.w3schools.com/jquery/jquery_ajax_load.asp
E. https://www.w3schools.com/jquery/jquery_ajax_get_post.asp

In addition, to get deeper explore JQuery's official site https://jquery.com.

End Notes

---

[a] This Lab manual bases on W3chools: https://www.w3schools.com

[b] W3schools Try it : https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_lib_google

[c] W3schools Try it: https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_on_multiple

[d] W3schools Try it: https://www.w3schools.com/jquery/jquery_callback.asp

[e] More at W3schools: https://www.w3schools.com/jquery/jquery_chaining.asp

[f] More at W3schools: https://www.w3schools.com/jquery/jquery_dom_set.asp

[g] More at W3schools: https://www.w3schools.com/jquery/jquery_dom_add.asp ,

   https://www.w3schools.com/jquery/jquery_dom_remove.asp

[h] More at W3schools: https://www.w3schools.com/jquery/jquery_dom_add.asp

[i] More at W3schools: https://www.w3schools.com/js/tryit.asp?filename=tryjs_cookie_username