Midterm Test Information

Please note the following information for the midterm test (25% of the final grade) scheduled for 22 Jun 2021, Tuesday between 8 AM – 5 PM. You will be able to attempt the test anytime between the duration mentioned above.

- The test will be delivered through SLATE quiz using Lockdown Browser with Respondus Monitor (Webcam).
  - Quiz having MCQs, Blanks, Ordering, Short Answers, and Matching question formats.
  - 2 hours duration
  - Closed Book
  - Only one attempt is allowed.
  - You will be presented 1 question at a time.
  - You will NOT be allowed to visit the previous question.
- All the topics covered in class, posted on SLATE and included in assignment from week 1 to 5
  - Reference sheet allowed having one sheet of paper handwritten or printed on both sides.
  - Please keep your OneCard or piece of photo ID handy as it will be required during the identification procedure before you can start the quiz.

Connectivity Issues:
During the exam, if you get disconnected from the internet and you are forced to exit the lockdown browser due to that, you are to take a screenshot/picture of that immediately and send it to me through Teams message. Then, try to reconnect the internet and restart your test. If any problems restarting the quiz, send me a message through Teams. I recommend you use a mobile app for the Teams.

Accommodation:
If you are entitled to extra time accommodation and want to use it for the test, please let me know latest by 18 Jun 2021 so that I can set that up for you.

Midterm Test Review

Section #1 : jQuery Selectors & Actions

Consider the HTML page given below:

```
<!DOCTYPE html>
<html>
<head>
   <title>Midterm Exam Review</title>
   <meta charset="utf-8">
   <meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1" />
   <script src="https://code.jquery.com/jquery-3.6.0.min.js" integrity="sha256-
     /xUj+3OJU5yExlq6GSYGSHk7tPXikynS7ogEvDej/m4="
     crossorigin="anonymous"></script>
   <script src="js/index.js" defer></script>
</head>
<body>
   <h2>Midterm Exam Review</h2>
   <!-- Section #1 : jQuery Selectors and Actions -->
   <h3 id="schoolName"></h3>

   <p class="myCampus"></p>
   <hr>
   <br>
   <section class="rhyme">Mary had a little lamb whose fleece was white as snow</section>
   <section>Little Bo Peep has lost her sheep</section>
   <section class="rhyme">Twinkle Twinkle Little Star</section>
   <br>
   <hr>

   <button id="toggleRhyme">Click to toggle element with class rhyme</button><br><br>
   <button id="hideStar">Click to hide the  element with class rhyme that contains
     "Star"</button>
</body>
</html>
```

```
/*
    Q-1: Write a jQuery code to include “Sheridan”  as the content in the element having id
    “schoolName".
*/

        $(“#schoolName”).html(“Sheridan”);
```

Assumption: assumed that the content has to be set and not appended

```
/*
Q-2: Write a jQuery code to include your main campus
as the content in the element “myCampus".
*/

$(“.myCampus).html(“Trafalgar”);

/*
    Q-3: Write a jQuery code to toggle the appearance of elements having class "rhyme"
on a click of a button “toggleRhyme”
*/

$(“#toggleRhyme”).on(“click”, function(){
        $(“.rhyme”).toggle();
});

$("#toggleRhyme").on("click", function()=>$(".ryhme").toggle());

/*
Q-4: Write a jQuery code to hide the elements having class "rhyme" containing "Star" on a
  click of a button having ID hideStar
*/
$(“#hideStar”).on(“click”, function(){
        $(‘.rhyme:contains(“Star”)’).hide();
});

/*
Q-5: Write a jQuery code to get the content of #schoolName element, store in a variable.
  Display the variable in the console log using the back ticks.
*/
        var school = $(“#schoolName”).html();
        consol.log(`SchoolName : ${school}`);
```

Section #2 : Local Storage

Consider the HTML code given below:

```
<!-- Section #2 : Local Storage -->
  <div>
    <h1>_____</h1>
    <!-- Enter your name on the line above -->

    <h2>_____</h2>
    <!-- Enter your student number on the line above -->

    <section>
      <label for="cname">Computer Make: </label>
      <input name="cname" type="text" id="cname" value="">

      <label for="ccost">Computer Cost: </label>
      <input name="ccost" type="text" id="ccost" value="">

      <button class="ui-btn" id="submitData">Submit To Local Storage</button>
      <button class="ui-btn" id="getData">Get Data from Local Storage</button>
    </section>

    <section>
      <p id="makeOutput"></p>
      <input type="text" name="cost" id="cost">
    </section>
  </div>

/*
Q-6: Write a jQuery code to enter your name and student number in the HTML area where indicated.
*/
$(“h1”).html(“JK”);
$(“h2”).html(“998989890”);

/* if both elements would have been h1 */
$(“h1:nth(0)”).html(“JK”);
$(“h1:nth(1)”).html(“998989890”);

/*
Q-7: Write a jQuery/JavaScript code to save the inputs given by user for computer make and computer cost to local
storage on the click of a submitData button.
*/



$(“#submitData”).on(“click”, function(){
```

```
	localStorage.setItem("computerMake", document.getElementById("cname").value);
	localStorage.setItem("computerCost", document.getElementById("ccost").value);

	//localStorage.setItem("computerCost", $("#ccost").val());
});


/*
Q-8: Write a jQuery/JavaScript code to retrieve the computer make and computer cost
from local storage on the click of a #getData button
and display on #makeOutput and #cost elements on HTML respectively.
*/

$("#getData").on("click", function(){
	$("#makeOutput").html(localStorage.getItem("computerMake"));

	// incorrect - $("#cost").html(localStorage.getItem("computerCost"));
	document.getElementById("cost").value = localStorage.getItem("computerCost");
});
```

Section #3 : JSON Retrieval

Consider the JSON data given below. Assume that the data is saved in the file named as **cars.json** within **jsonData** sub-directory in the project directory.


```
{
   "information": "Sheridan Cars",
   "cars": {
      "New": [{
            "make": "VW",
            "type": "Bug",
            "engine": "gas",
            "cost": 24000,
            "colors": ["red", "blue", "silver"]
         },
         {
            "make": "GMC",
            "type": "Suburban",
            "engine": "diesel",
            "cost": 18000,
            "colors": ["silver", "brown"]
         }
      ],
      "Used": [{
            "make": "VW",
            "type": "Bug",
            "engine": "diesel",
            "cost": 12000,
            "tag": false,
            "colors": ["red", "blue", "silver"]
         },
         {
            "make": "GMC",
            "type": "Suburban",
            "engine": "diesel",
            "cost": 10000,
            "tag": true,
            "colors": ["silver", "brown"]
         },
         {
            "make": "FORD",
            "type": "F150",
            "engine": "diesel",
            "cost": 20000,
            "tag": true,
            "colors": ["silver", "black"]
         },
         {
            "make": "HYUNDAI",
            "type": "Elantra",
            "engine": "gas",
            "cost": 10000,
            "tag": false,
```

```
            "colors": ["silver", "brown"]
        }
      ]
    }
}
```

Consider the HTML code given below:

```
<!-- Section #3 : JSON Retrieval -->
   <div>
      <section>
         <h3>All Used Diesel Cars with Tag Included</h3>
         <table>
            <tr>
               <th>Make</th>
               <th>Type</th>
               <th>Cost</th>
            </tr>
            <tbody id="jsonbody">

            </tbody>
         </table>
      </section>
   </div>

/*
Q-9: Write jQuery code to make initiate a request to get the JSON data from cars.json
file from jsonData sub-directory within current directory.
If successful, call method name fetchCars. Otherwise, display alert with appropriate message.
*/
   $.ajax({
            type: “GET”,
            url: “jsonData/cars.json”,
            datatype: “json”,
            success: fetchCars,
            error: function(request, error){
                        alert(“Unable to fetch data ” + error);}
            });

/*
Q-10: Write jQuery code for fetchCars method assuming a function parameter receiving json data from AJAX call.
Using the received data, extract the used car objects, and  iterate through all the extracted used car objects to display
the object details in the log message on console.
*/
function fetchCars(data){
            let usedCars = data.cars.Used;

            for(car of usedCars){
                        console.log(car);
```

```
		console.log(`Make : ${car.make}`);
	}
}

/*
Q-11: Write jQuery code for displaying the individual table rows in #jsonbody table
for each of the used car object you extracted and iterated over in the previous answer. Check the table header for the details you will need to display in a table row.
*/

function fetchCars(data){
	let usedCars = data.cars.Used;

	for(car of usedCars){
		$("#jsonBody").append(`<tr><td>${car.make}</td><td>${car.type}</td><td>${car.cost}</td></tr>`);
	}
}
```