American University of Beirut
Department of Computer Science
CMPS 278 – Web Programming

Spring 2022/2023

Asst 4: Due on Mar. 2 at midnight

Problem 1

Create a webpage that contains 3 forms according to the following (none of them is submitted anywhere):

1) **Form 1**: this form uses a function that is expected to be in some external JS file linked to your page. The function determines whether a given year is a leap year in the Gregorian calendar or not. Your form should contain a *label* reading "Year", an *input* of type text, a *textarea* whose id is "*result*", and a button labeled "Check year". When the button is clicked, you should call the function and pass to it whatever year the user entered. The function should then return whether the year is leap or not. Accordingly, you should append to the text area's content the sentence "**[yyyy] is a leap year**" or "**[yyyy] is not a leap year**", where [yyyy] is the year entered by the user. If the text field has no value or it is not composed of four digits then you need to append to the text area's content the following statement written in capital letters "THE ENTRY IS INVALID". You should preserve the content of the text area at all times i.e. the content should not be overridden but rather appended whatever the result is after every click of the button.

2) **Form 2**: add another form to your page that allows a user to validate a credit card number. The latter should be entered in a text field assuming that it has the following format "dddd-dddd-dddd-dddd" (where d is any digit). The form should contain a button labeled "Validate", and a <p> element whose id is "result". Once clicked, the button should call another JavaScript function written in the same file used by the Form1 to validate the credit card's number. The function should return `true` or `false`. You should update the content of the sentence with "Card number dddd-dddd-dddd-dddd is valid" in green in case it is valid and with the sentence "Card number dddd-dddd-dddd-dddd is invalid" written in red and make sure to mention the reason(s) as shown below. To tell whether a credit card's number is valid check the following:

    - Number must be composed of 16 digits.
    - You must have at least two different digits (i.e. not all of the digits can be the same)
    - The final digit must be even
    - The sum of all the digits must be greater than 16

    Example, the following credit card numbers are valid:

    - `9999-9999-8888-0000`
    - `6666-6666-6666-1666`

    whereas, these are not:

    - `a923-3211-9c01-1112` *invalid characters*
    - `4444-4444-4444-4444` *only one type of number*
    - `1111-1111-1111-1110` *sum less than 16*
    - `6666-6666-6666-6661` *odd final number*

3) Create, yet a third form that can be used to simulate your final grade for CMPS278. The page should allow the user to enter the grades of all the homework (in a single textbox), the project, all the quizzes (again, in a single textbox), the attendance, and final exam as shown below. Assume that the grades' weights are as follows:

Assignments	10%
Project	15%
QUIZ1	15%
QUIZ2	25%
Final exam	30%
Attendance	5%

It is left to you to use the appropriate HTML elements to provide the above (the form should not be submitted anywhere. All calculations and output are on the same page as the previous two forms).

We assume that all the fields are initially set to 0.

A function should get the values, do the math, and then show the result accordingly.

After entering/updating the grades for the quizzes, your script should compute and display the quiz average, and update the course overall average and the raised grade.

Any time one of the grades is updated, the function should be called to reflect the change.

You may accept (again, in the same input element) at least four grades for the assignments, up to six at max. There has to be exactly 3 grades for the quizzes (again, in the same input element).

Note that the final grade is a weighted average i.e. you need to multiply each component's grade by its percentage.

Problem 2

Add a Javascript to add behavior to the following HTML code. Use the set of icons found in folder "p2".

```
<div>
    Name of Food: <input id="foodname" type="text" />
    Food Group:
    <select id="foodgroup">
        <option>fats</option>
        <option>fruit</option>
        <option>meat</option>
        <option>veggies</option>
    </select>
    <button id="eat">Eat!</button>
</div>
```

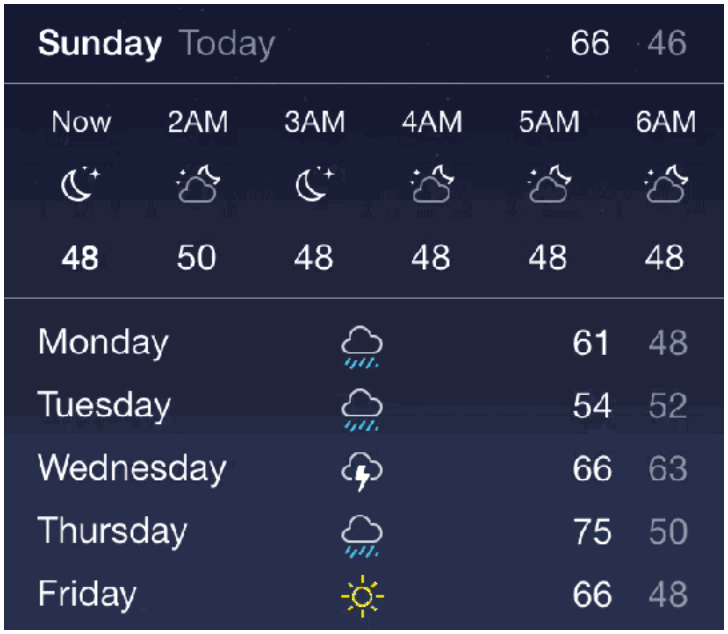
```
    <p>
        <img src="cookie.jpg" name="cookie" alt="fats" />
        <img src="apple.jpg" name="apple" alt="fruit" />
        <img src="broccoli.jpg" name="broccoli" alt="veggies" />
        <img src="tomato.jpg" name="tomato" alt="fruit" />
        <img src="drumstick.jpg" name="steak" alt="meat" />
        <img src="milkjug.jpg" name="milk" alt="dairy" />
        <img src="potatochips.jpg" name="chips" alt="fats" />
    </p>
```

The page contains a text input box with `id` of `foodname`, and a select with an `id` of `foodgroup`. The user types a name of a food item into the `foodname` text box, such as `apple` or `Cookie`, selects a food group from the select element, such as `dairy` or `fruit`, and then clicks the button whose `id` is "`eat`". When the `eat` button is clicked, any element on the page that matches the specified criteria will be hidden. Mind the following:

- Only elements of type `img` should be hidden.
- The food item's `name` should match that value entered by the user in the `foodname` box. The food's name is stored as the image's `name` attribute. For example, if the user types `cookie`, only `img` elements with the `name` "cookie" are hidden. Your code should be case-insensitive; for example, `coOKIe` affect all images whose names spell all forms of the word "cookie".
- In addition to matching the name of the food item, the images to be hidden should belong to the group specified by the user in `foodgroup` list. Food groups are represented using the `alt` attribute. For example, a jug of milk would have the following element:

```
<img src="milkjug.jpg" name="milk" alt="dairy" />.
```

Problem 3



Your job is to create a page that uses HTML, CSS, and JavaScript to generate a similar page. To do this, you need to use the set of icons packed with this homework in folder "p3". Your script should find dynamically what day is today and then generate randomly a weather forecast for one week starting today. In the image shown above, it is assumed that it is a Sunday. Therefore, you should generate a forecast for the weather from Sunday till Saturday (Saturday is not shown in the image above). Next to each day, you need to show an icon picked up at random and two temperatures (in ˚C): the lowest and the highest (assume that the range of values is [-12, 40]) aligned horizontally.
In addition, and on top of the week's forecast you need to show today's weather forecast on an hourly basis. Again, you need to show for every hour a randomly generated icon, a lowest temp and a highest temp aligned vertically.

You will submit via Moodle a compressed file called **AUBnetAccount-hw3.zip/rar**. The compressed file should contain two subfolders named problem1and problem2. Put in these folders all of the files related to your solution (HTML, CSS, JS, images, etc.).

Please do not place a solution to this assignment online on a publicly accessible web site. Doing so is considered a violation of the academic integrity policy.