EECS 1012, Lab 8 – More on JavaScript
November 11–November 16, 2019

A. REMINDERS

1) You **must** attend your **assigned** lab session (we will be marking your submission in the lab).

2) You must arrive on time: anyone later than 15 minutes may not be admitted to the lab.

3) You must complete the pre-lab mini quiz posted on Moodle in the first 15 minutes of your lab time.

4) Each lab including the pre-lab mini quiz is about 2% of your overall grade.

5) TAs are in the lab to help you. They will also verify and mark your work at the end. Signal a TA for help if you stuck on any of the steps below. Yet, note that TAs would need to help other students too.

B. GOALS & OUTCOMES FOR THIS LAB

To practice more concepts in programming, including variables, arrays, functions, and program control statements

To use JS objects, such as document, Math, and Date

C. TASKS

1) TASK 1: Simple button generating random output using an if-statement.

2) TASK 2: Passing string variables to functions.

3) TASK 3: Passing numeric variables to functions.

4) TASK 4: Random + string concatenation + global variables + for-loop.

5) TASK 5: Date object + array + string concatenation.

6) TASK 6: Global variable and if-statement

INCLUDED WITH ZIP FILE: The zip file contains an example (in the example folder) that uses random, arrays, functions, and an if-statement.

This zip file also contains a video

finding_JS_errors_no_audio.mp4 – which shows how to use the Browser's console to help debug JS errors. Note that when using Firefox, the equivalent functionality is available via Web Console (Ctrl-Shift-K). Try to find an error in the included example.js file (in the example folder).

SUBMISSIONS

1) Manual verification by a TA

You will need to have one of the TAs of your lab to verify your work before submission. The TA will look at your various files in their progression. The TA may ask you to make minor modifications to the lab to demonstrate your knowledge of the materials. The TA will mark your name off a list; You are required to sign the list to show your attendance and that you have been verified.

2) Moodle submission

You will see an assignment submission link on Moodle.

1) Create a **folder** named "**Lab08**" and copy **all** of your HTML and JS files. This folder should be compressed (or tar.gz), and the compressed file submitted.

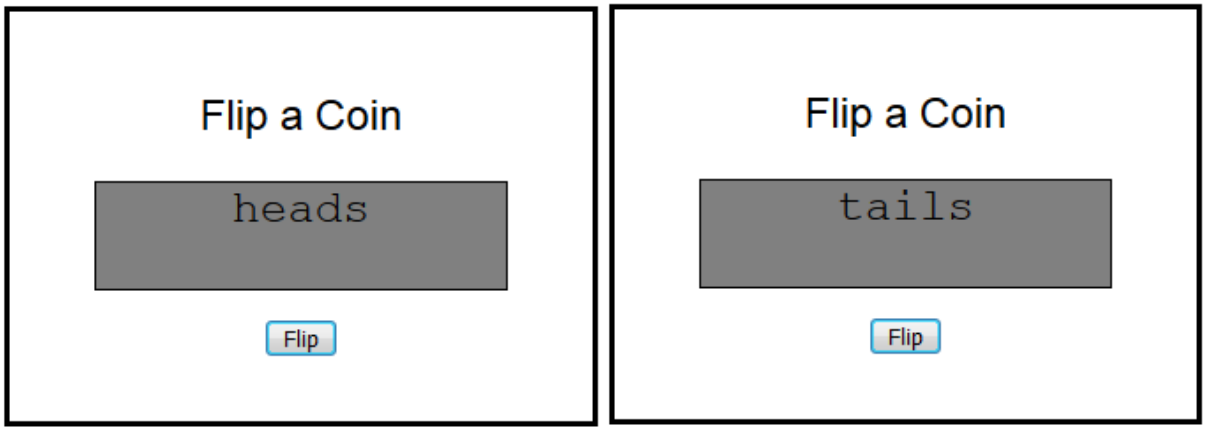
Task 1: Edit task1.js (you do not need to edit the HTML file).

For this task, we have already declared the JavaScript function `myFunction()` for you.

Your function should do the following.

Each time the button is clicked, your `myFunction()` code should generate a random number. If the random number is greater than or equal to 0.5, then have the `innerHTML` of the paragraph variable set to "tails", otherwise set it to "heads".

See below for example outputs.



Task 2. Edit task2.html and task2.js

(1) Link your task2.js to your HTML code.

(2) Have the text in the paragraph "mydata" start with **Value** (see below).

(2) Add four buttons to your Task2.html as shown below.

(3) Write a function in JavaScript that has one parameter. When a button is pressed, it should pass the value shown in the button (e.g., "Toronto", "Montréal", …). Your function should change the innerHTML of the paragraph to the passed value as "Result = **VALUE**". See example below.

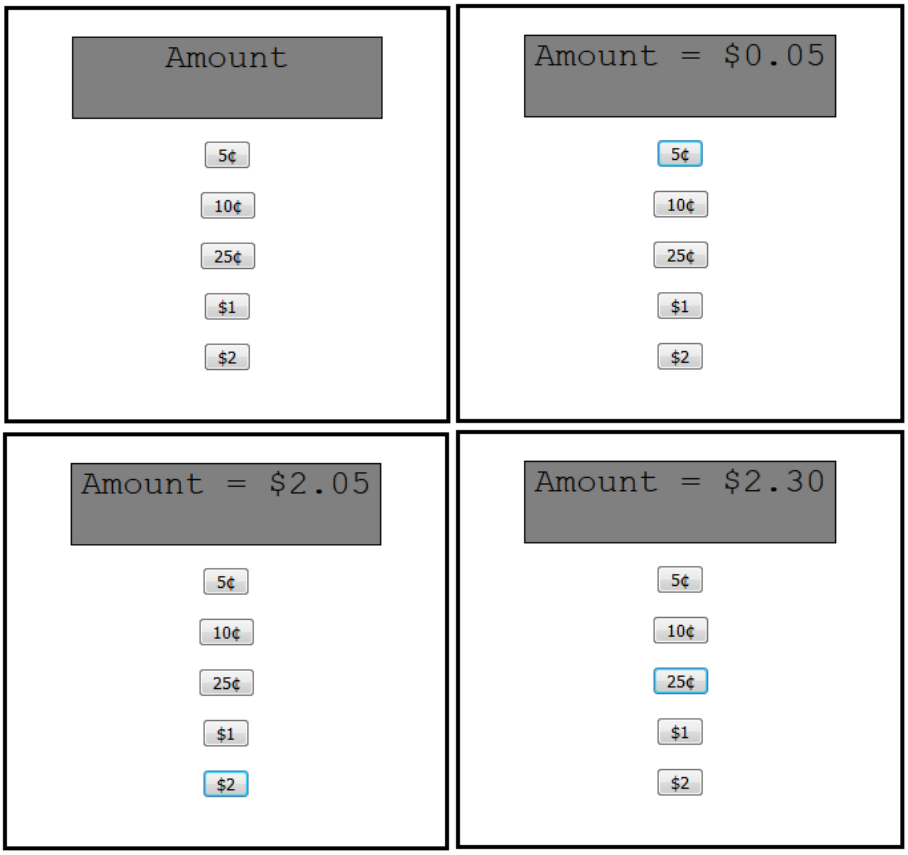
Task 3. Edit task3.html and task3.js

(1) Link your task3.js to your HTML code.

(2) Have the text in the paragraph "mydata" start with **Amount** (see below).

(2) Add four buttons to your Task3.html as shown below.

(3) Write a function in JavaScript that has one parameter. When each button is pressed, it should pass the *value* shown in the button (e.g., 0.05, 0.10, 0.25, …). The function should add the value to the existing one – as illustrated below. For that, you might want to use a *global* variable[1].



[1] Global variables are variables that are accessible by *any* function inside a JS file. They are placed outside any functions you might have there. More on global variables in Task 6.

Task 4. Modify task4.html and task4.js

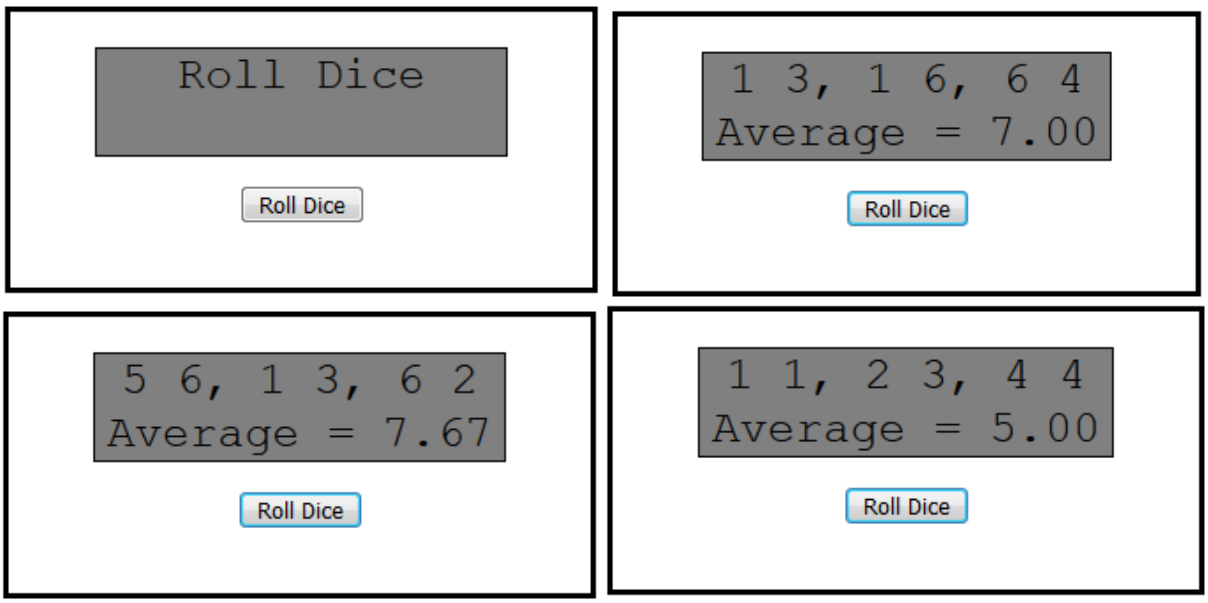
(1) Link your JavaScript file to your HTML file.

(2) Have the text in the paragraph "mydata" start with **Roll Dice**. Add a button "Roll Dice". Have this button respond the click event.

(3) Have the onclick for your button link to your JavaScript function. The function does not have parameters.

(4) Each time you click, have your function compute **three sets** of **two** random numbers from 1 to 6. These represent three sets of dice rolls. When rolling the dice display the resulting values, as shown below. Use the *innerHTML* to present the die values. Note, there is no comma after the last set of numbers

(5) At the end, display the average value of the three tosses, as shown below



Task 5. Modify task5.html and task5.js

(1) Link your JavaScript file to your HTML file.

(2) Have the text in the paragraph "mydata" start with **Today's Date**. Add a button "Click". Have this button respond the click event.

(3) Have the onclick for your button link to your JavaScript function. It does not have parameters.

(4) When you click, your function should create a Date object.

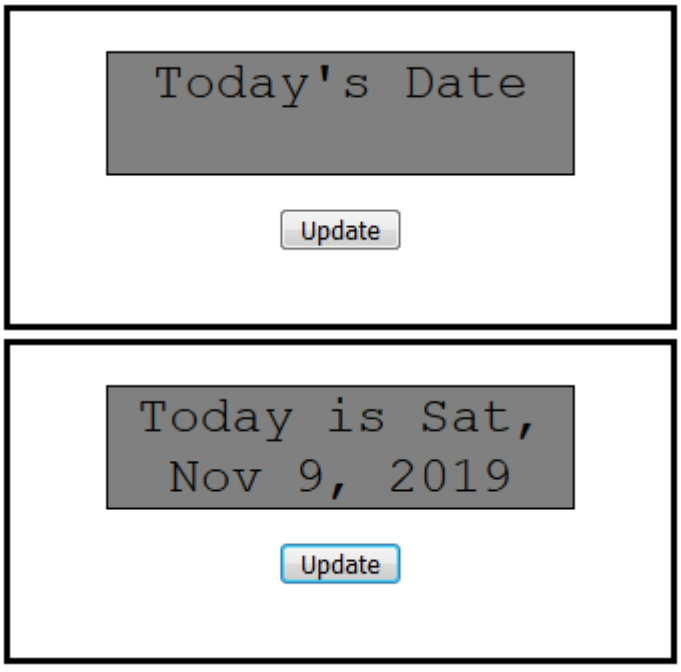
Get the following data from the Date object[2].

(i) day of the month
(ii) day of the week
(iii) month
(iv) year

Using this data, change the innerHTML to output the string below.

Hint: You should use an array to store the three letter days of the week ("Sun", "Mon", "Tue", …).

Hint: You should use an array to store the three-letter abbreviation of the month ("Jan", "Feb", …).



[2] https://www.w3schools.com/js/js_date_methods.asp

Task 6. Modify task6.html and task6.js

(1) Link your JavaScript file to your HTML file.

(2) Have the text in the paragraph "mydata" start with **Count Down**. Add button "Click". Have this button respond the click event.

(3) Declare a global variable. This is a variable that is created outside your function. Inside your function, you do not need to declare it again. If you modify the variable, the modification will be remembered next time you access the function. See example code here.

```
var i=20;
function myFunction()
{
 i = i--; // the value of i will be remembered next function call
}
```

(4) Each time your button is clicked, you should print out the global variable and reduce it by 1 (one).

Your innerHTML of the paragraph with id "mydata" should show the current value of the global variable.

(4) When the variable gets to 0 or less, have the your innerHTML change to **liftoff!**

See on side.