JavaScript Lab 1

***At this point you should understand the following:***

***(There is a difference between recognizing and understanding!) (Watch the Team Tree House and other Videos before attempting this lab in case you’re not familiar with any of the following)***

1. Know: the difference between document.write and document.getElementById.
2. Know what you can change using document.getElementById
3. Know what the innerHTML is
4. Know how to use all of the above.
5. Know how to write out variables using both innerHTML and document.write
6. Know how to create a variable
7. Know how to create an array
8. Know how to create a prompt box, and how a variable holds what the user types in when a prompt box pops up
9. Know how to create a confirm box, and how a variable holds what the user clicks on in response to the confirm box.
10. Know how to generate a random number.
11. Know how to use the random number to access something in an array
12. Know how to create an if condition
13. Know how an else if and an else works

***And this week you learned:***

1. Know how to write a function
2. Know the different ways to call a function (make a function happen)
3. Know how to go sequentially through an array
4. Know how to figure out how many items are in an array, and how to use that.

# Lab

**Debugging hints:**

* 1. If nothing is working, there’s probably a typo you’ll have to find.
     1. Look for capital letters where there should be a small one, and vice versa
     2. Look for missing “ ” (if you open it, you must close it)
     3. Same with { and }
     4. Same with ( and ) – for every one of the first, you must have one of the second.
  2. Use comments /\* \*/ to isolate what is working and what isn’t. By using comments, you can figure out exactly where you need to focus to find the bug.
  3. Finally, make sure your html is valid (again, if you opened a tag, you should close it)

# Problems (100 pts):

***(Note: On your web page you can have something like,***

***<img src = “cat.jpg” id = “img1” width = “300” height = “400” alt = “cute cat”>***

***When I refer to the image on your web page, I’m referring to the <img…> tag, whereas when I refer to the picture, I’m talking about what the src is set to, or “cat.jpg”)***

1. (3 pts)Create a web page with a paragraph with an id of “p1” that says, “Secret Message”. The paragraph should have a call to a function using onMouseOver. The function (placed in a script in the head section of your web page) should change the paragraph p1’s innerHTML to “Wombats is Stabmow backwards”. When you move your mouse off of the paragraph, the innerHTML should change back to, “Secret Message”
2. (3 pts) Create a web page (or modify the existing web page) with an image on it of some picture of your choice (we’ll call this picture1.jpg). Write a function that uses getElementById to change the picture in that image to something new (picture2.jpg). Write a second function that changes the picture in that image back to the first picture (picture1.jpg). Now modify the image on the page so that when you run your mouse over it, the first function is called, and when you run your mouse off of it, the second function is called.
3. (7 pts) Create a web page (or modify the existing web page) with a header, some paragraphs, and an image on it. Add a button that says, “Click here for better accessibility”. Make the button call a function. The function should change the style of the page to adapt to the needs of people with low vision (e.g., bold font, large font, arial font, background black and font color white, make the image(s) on the page larger, and make the image have a longer alt tag ).
4. a. (3 pts) Create a web page (or modify the existing web page) with an image on it, a paragraph on it, and a button. Now create a script (in your head section) that contains an array of pictures and a variable initialized to -1. It should also contain a function that will first increase the variable by 1, then check to see if the variable is longer than the number of elements in the array, and, if so, resets the variable to 0. The function then displays the picture in the array at that variable number. Now make the button on your web page call that function (note – this is pretty much what we went over in class). Make sure that if you add pictures to your array, this function will work regardless of how many pictures you add.

4 b. (5 pts) Now add another button and another function. This function should allow you to add pictures to your array. The second button on your web page should call this second function. (again, this is pretty much what we went over in class).

4 (6 pts) c. Now, inside your script, but above your functions (either above or below the first array of pictures), create a second array. The array should hold text describing (in order) each of the pictures in the array of pictures you created. (Remember to put quotes around the text, so the array will look something like this:

textarray = new Array()

textarray[0] = “description of picture 0 goes here”

…

Now modify the function you wrote in 4a so that it also changes the paragraph on the web page’s text to what is in your array of text at the variable

4 (7pts) d. Finally, modify function 4b so that when you add a picture to your picture array, you must also add text describing the picture to your text array (you’ll use two prompts for this: the first will get the new picture, and the second will get the new text).

4 (7 pts) e.: Add a paragraph to the web page. Then modify function 4a so that it will tell you, “Picture 3 of 5” where 3 will be replaced with the picture’s number, and 5 will be replaced with the number of pictures in your array.

1. (13 pts) Write a web page (or modify the existing web page) with 4 images on it (two for your dice, and two for the computer’s dice). The page should also have a paragraph on it (it can start out saying something like, “see how you do against the computer”). It should also have one button on it, saying “Click here to roll the dice”. Clicking on the button should call a function.

Now inPi the head section write a javaScript The javaScript should have an array of 6 pictures, each picture being a different face of a die (the singular of dice- you can use the pictures you downloaded for js lab1).

The javaScript should also have a function. The function should generate 4 different random numbers, each between 0 and the length of the array. The first random number will cause the picture in the array at that number to be displayed in the first image on the page. The second random number will cause the picture in the array at that number to be displayed in the second image in the page, the third random number will cause the picture in the array at that number to be displayed in the third image on the page, etc. for the fourth image.

Assume the first two images (and hence the first two random numbers) are your dice roll, and the third and fourth images (and hence the third and fourth random number) are the computer’s dice roll. You need to figure out and print out who won using the following criteria:

If either of the computer’s dice is a 1, the computer loses and you win (and that should be printed out in the paragraph on the page).

Otherwise, if either of your dice is a 1, you lose and that should be printed out to the paragraph on the page.

Otherwise, if your total dice roll (the sum of both your dice) is greater than the total dice roll of the computer, you win, whereas if the computer’s total dice roll is greater than your roll, the computer wins, whereas if you both roll the same amount, it’s a tie. This should be printed out accordingly in the paragraph on the page.

Remember, the array should go from 0 – 5, but the die pictures will go from 1-6. This means that if you want to check to see if either of the random numbers is a 1, you would say the following:

if (randnum3 == 0) || (randnum4 == 0)

{…

}

|| means or in javaScript.

1. (13 pts) Write a web page (or modify the existing web page) with an image and a corresponding paragraph. Create a button below the image and paragraph, and then a second paragraph below the button.

Now have the button call a function.

The function should contain 2 equal-length arrays: An array of pictures of faces displaying emotions, and a corresponding array of sentences describing the emotion. Then the function should generate 2 different random numbers. The first should cause the picture at that number in the array of pictures to be displayed in the image on the page. The second random number should cause the sentence in the array of descriptions to be displayed in the images corresponding paragraph on the page.

A confirm box should then ask the user, “Do these go together?”

If the user clicks “ok” and the two random numbers match, you should write to the third paragraph, “Good job! You are correct.” If the user clicks “ok” and the two random numbers don’t match, you should write out, “Sorry, that is not correct.”

If the user clicks “cancel” and the two random numbers match, you should write, “Sorry, you’re wrong. These two do match”, and if the user clicks “cancel” and the two random numbers don’t’ match, you should write to the paragraph, “Correct, these two don’t belong together.”

(Note: this could be a training exercise for children with autism. If you want to use this methodology to create a training tool, for, say, a foreign language (with pictures of something and a description in a different language), or anything else, you can. I don’t care what the array of pictures and the array of sentences contain, just that you create the training tool).

***Note:*** *If, for some reason this doesn’t work in Chrome, try running it in Firefox. Chrome is funny about when it actually executes statements (there’s a way around this that we’ll learn shortly). In the meantime, just run it in Firefox.*

1. (8 pts) Write a web page (or modify the existing web page) with an image on it. Position the image absolutely, with the left position being at 0 px. Add a button somewhere lower in the page that calls a javascript function using onclick.

Now add a javaScript. The javaScript should have a count variable, just like we had for going through an array sequentially. It should also have a function. Inside the function, you should first increase the count not by 1 (as we did for the arrays), but by 10. You should then use document.getElementById’s style properties to change the position of the left to be count pixels over. (You’ll have to use say:

document.getElementById("imgid").style.left=count+"px";

where “imgid” should be replaced by the id of the paragraph the image is in on the page (or the image itself, if you’ve styled the image instead of a paragraph surrounding the image).

Now when you click on the button, the image should move to the left by 10 pixels each time.