Unit 4 Learning Diary

Learning Javascript

I used this course from Rob Dey to learn the fundamentals of Javascript. I learned other elements on the web through simple web searches and then visiting the sites that came up. Obviously, I'm not writing fully-fledged scripts yet – that's for next unit. Right now, I'm just focused on knowing the fundamentals and being able to identify good javascript from bad javascript.

Distinguishing Good from Bad Javascript

I went to **javascripts.com** to see if I could find examples of good/bad javascript code.

Here is an obviously bad example of bad Javascript:
Example 1 If you visit this page, you're provided with a script as well as some CSS and HTML. I'm not concerned with the CSS or HTML, but the javascript. If I copy/paste the given code into my own file, I can an unreadable mess. This is as bad as it gets.

Example 2 This is an extremely short script. It's unambiguous, but the creator demonstrates its use within an HTML document. Putting your javascript inside the HTML document isn't a good practice. It's obtrusive code; HTML is for describing the structure of a page, not its behaviour.

Example 3 This script isn't inside any script tags. The functionality comes from "onclick = function()" lines of code within the HTML elements. This is another example of obtrusive code. The best way to incorporate Javascript is to have it in a separate file completely, and link to that file from a line in the section of your HTML document.

Example 4 (Cups) This script come from Codepen. Javascripts.com is a good site to find a lot of bad javascript. Codepen is a good site to find well-structured javascript along with its HTML and CSS. The example here shows Javascript being used to animate 3 rotating cups. The script is well-indented and no longer than it needs to be. I include part of the downloadable zip-file in my portfolio for reference. Unfortunately, there aren't any comments. Good Javascript has comments where appropriate. Over-commenting is also a bad thing.

Implementing Javascript

I have several ideas of code snippets to use in my site. First, I want to validate the forms on the Contact page. Second, I want the cards on the Products site to be flippable upon a **click**, not a **hover** event. And… yeah, that's pretty much it. Thankfully, the site looks nice already, and with the kind of pages I'm building, they won't need much Javascript. For my own javascript in the next unit, I want to incorporate a feature on the Development page which enables you to zoom in on the photos there, and then collapse them again. For JQuery (comes Unit 6 I think), I'll just put some click drop-downs on the home page. For the last unit, I'll put up a twitter feed on the Contact page, and maybe something to do with Turri's other social presences on the web.

Right off the bat, I had problems. I couldn't get any javascript to run effectively when I put it in a separate file. If I put the identical code in **< script >** tags in the HTML document, it did work. After doing some research, I found the solution on this site from Stack Overflow, where I found a solution to wrap all my JS code inside **window.onload = function()** {*my script here…*}

I decided to start with the forms on the Contact page. I found this cool page on w3schools.com that showed a simple validation script. All it does is see if a form is empty when you press "submit", and if so, it doesn't actually submit the information and sends an alert that says "so-and-so must be filled out." This script only works for a form at a time, so I modified it to work for all the forms on my contact page. I created 7 variables within the function, one for each form. If any of them is empty, the information isn't submitted. I also added the email validation code from this site as part of my validateForms() function.

I implemented this javascript with potential customers in mind. To be specific, personas 1 and 3; also, Scenarios 2, 4, and 5.

My second project was making the cards in the Products page flip on click rather than hover. I encountered a slight problem here. Originally, I had used the tutorial in this video to make the flip cards. This teacher didn't explain how to make the cards flip on a click event, so I had to find another tutorial. I found this video, which did explain what I wanted to know. The thing was, this coder's flip cards were based on different HTML and CSS than the previous coder's. I tried to do what he said, but the code wouldn't execute properly and I was getting errors in the console. So, I simply re-programmed my cards according to his code (this required adding new classes to my HTML and restructuring the CSS), after which the code executed properly.

There was still a slight problem. This coder only had 1 card which he added an **event listener** to. This event listener watched for an event – in this case, a click – and upon the click, executed some code (in this case, the card flipping). I had 15 cards on my page!! Naturally, I added event listeners to all of them. This resulted in a lot of (what I thought) should be unnecessary code. There must have been some way to iterate through all the relevant HTML elements and add event listeners to them, without manually typing them all out.

I tried to do this with a standard **for loop** – for(var i=0; i<array.length; i++){}. It wouldn't work – it would only add an event listener to the last card in the sequence. This was pretty frustrating, and I spent a long time looking for solutions online. I tried a lot of different stuff – do…while() loops, if loops with

variables declared beforehand, forEach functions. None of them worked. Then I came across this lifesaver, which offered a solution to use a slightly different for loop:

for(**let** i=0; i<array.length; i++) {}

The only difference was that I used **let** instead of var. This gave the variable [i] “block scope”, which allowed the loop to do the same thing for each element in the array without overwriting certain things. Now, I had the proper event listeners on all my cards without repeating the same code for 15 different cards and cluttering my JS.

I implemented this javascript with potential customers in mind. In particular, persona 1 and scenario 2 (info can be found in my Unit 1 diary).

Links to resources I used/benefited from

Where I got the idea for an email validation script

Basis for most of my form validation script

Source of extra email validation scripts I used

How to iterate over HTML elements and add event handlers to each

Where I learned about the underlying cause of external JS files not working

Side-notes

I also added a small change to all the link buttons on all the pages within the site. If you click any of them, they will appear white. When idle, they’re red; when being hovered over, they’re green; when clicked, they’re white. I managed to incorporate all the colors of the Italian flag onto them, which is fitting, considering the company this site is representing.