Rissa Lee
PUI Section B – Assignment 6B

Reflection

I ran into a lot of bugs and issues as I worked on assignment 6B, but the process of having to develop and debug my Javascript code really helped me better understand the language and object-oriented programming on a conceptual level. Throughout the entire process, I relied a lot on the Javascript console in Google Chrome to help me figure out what I was doing wrong, a skill that was emphasized and taught during lab.

Some of my most common bugs had to do with accessing the wrong element. For example, I created a cart array and then added Bun objects to the array whenever users added buns to the cart. At first, when I was trying to access the properties of specific bun objects within the array I kept running into errors and seeing "undefined" in the Javascript console. I realized after some time that I had not been specifying which element within the array to look at. Once I properly accessed the object within the array, I was then able to see the Bun properties I had created earlier on in a Bun constructor.

Another common issue I ran into was keeping track of all the ids in my HTML files when using them in my Javascript file. For instance, I would try creating an id like "cartTotal" only to find out that I had already used that id somewhere else. As a result, I found myself creating a new lengthy camelCase id on the spot—"cartPageTotal." This made the process of debugging and coding a lot more confusing. In the future, I think it would benefit me to start with an id system for all the HTML elements that I would need to access later in my Javascript file; this way, I would have a document to reference on the side and I would avoid creating lengthy and descriptive ids that are similar to one another.

Over the past few weeks and especially the last week, I thoroughly enjoyed working on this assignment. It was very challenging, especially during the weeks that involved Javascript (I had never worked with Javascript before), but the process of having to debug my own code taught me my specific problem areas and how to overcome them. I also learned a lot about the capabilities of HTML, CSS, and Javascript and I now feel like I have a better understanding of what is feasible in a website design.

Programming Concepts

1. Data structures
    a. I learned that the use of constructors makes it easier to create the prototype of an object and apply its structure across a website. In my Javascript file, I created a constructor called “Bun” that allowed me to create a new bun any time a user added something to their cart. By using a constructor rather than writing out the object every single time, I saved myself a lot of effort and potential errors.
2. Variables
    a. I learned that variables can be incredibly powerful, especially when put into storage. For this assignment, I used *sessionStorage* to keep track of items users added to the cart on the PDP so that I could still access these variables when users moved to new pages. By storing variables like “cartTotal” (counter next to the cart icon) and “shoppingCart” (array of all added buns), I was able to dynamically update the cart icon counter and the list of items on the cart page even when users were looking at different parts of the website. I think that this really helped create a cohesive experience overall.
3. Syntax
    a. I learned that the syntax of Javascript is very specific and powerful. Even with a small typo like *document.getElementByID*, I ran into obstacles that often left me confused for a long time. On the other hand, when I did use the syntax properly, I felt like I was able to accomplish really powerful and dynamic abilities by combining Javascript with HTML and CSS. One of my favorite features I implemented was on the PDP where I checked to see what glaze had been selected and updated the image on the PDP to reflect that.
4. Tools
    a. I learned that the Javascript console in Google Chrome is a great tool to debug code. Every time something didn’t happen that should have happened on my website, I knew to open up the Javascript console and it would tell me where my issue was occurring. What I especially liked about the console was that it told me the exact line of code to look for the error. After a while, I got so used to using the Javascript console to debug my code that I kept it open all the time whenever I went to check my website in the Chrome browser.
5. Syntax/Tools

a. I learned that *console.log* is the best helper when trying to debug issues or keep track of my progress. For instance, if an issue was flagged to me in the Javascript console, I would go to the line where the error happened and *console.log* it to see what exactly was being accessed. Often times, I would find that I was accessing the wrong thing entirely. This was more of a reactive use of *console.log*. A proactive way that I used *console.log* was by calling it on values that were important to me but not necessarily visible on the website. For instance, if I added an item to the cart on the PDP, I wanted to know that it had updated the total price of the cart (not visible on the PDP). When I used *console.log* on that variable, in the Javascript console, I was able to check that it had updated without having to change to the cart page.