Michael Silvestre
PUI
10/30/19

Reflection and Programming Concepts

Reflection

The biggest obstacles I had with this assignment was working with local storage. I have never had to utilize that tool in JavaScript and trying to figure out exactly how it worked was an initial struggle. Ultimately, a combination of the lab slides, Stack Overflow, and Chrome's debugging tools allowed me to slowly make sense of the process. Once I was able to visually see how data was transferred through local storage via stringify and parse methods, I was able to understand how to manipulate the objects so I could populate the data on my tables.

Another challenge I had was working with event handlers. I always seem to struggle with getting them to function asynchronously when I'm coding. The easiest way I've found to work through this issue is by calling them within the HTML; however, I would prefer to learn how I might be able to utilize these without manipulating my HTML code and keep things cleaner within the "Model-View-Contoller" mindset of programming.

Programming Concepts

1. **Local Storage:** This is perhaps the biggest thing I learned from this part of the assignment. I might have vaguely heard about it once before, but I never had a chance to use it and see how it functions. Being able to store my objects within a value in localStorage allowed me to think more abstractly about my code and populate information across pages.
2. **parentNode:** Utilizing nodes has always been extremely confusing for me and I have admittedly tried to avoid them; however, they were absolutely essential in helping me program my remove function in my code. Being able to think about the DOM within these nodes was tremendously helpful in scaling my code and making these functions simpler to write.
3. **parseInt():** The parseInt() method was a useful tool for being able to manipulate elements of my HTML. I was able to extract the quantity property from my Bun object, which allowed me to do useful things on my website. This was really helpful for calculating price of each bun package. If I had more time, I was planning on implementing a total feature where it ran through each of the price cells and summed up the total.
4. **Mode slips:** This is a more abstract concept, but I wanted to make sure that my forms accounted for possible errors in mode. I wanted to make sure that the page had some kind of recognition that the information was sent to the cart beyond a visual indication of the number. How I went about solving this was resetting the dropdowns for quantity and selection so that they went back to their default states. That way, someone would not submit the same information over and over again.

5. **Debugging:** This is another abstract programming concept, but as I was debugging I encountered how my code could create usability issues. When I made the initial add to cart feature, I realized that It could have been problematic to allow people to submit information without all of the forms filled out. I created a conditional that recognized when quantity was not a number that helped prevent people from submitting product information unless these values were sufficiently filled.