HW6B Nicole Xiang
Host Link: https://nicole-xiang.github.io/PUI_HW6B/
Repository Link: https://github.com/nicole-xiang/PUI_HW6B

Please refer to navigation steps in the readme when browsing the website.

Reflection

Challenges

Some of the biggest challenges I encountered were 1) adding items to cart using localStorage and 2) showing items that are added to cart. Because I wasn't really familiar with localStorage yet, I was a little lost in the initial stages of the development of the add to cart feature. After reading some resources online and asking the TAs, I figured out my bugs and the best approach to this feature. Specifically, I had a difficult time with the onclick javascript function used by the add to cart form because the page kept refreshing whenever I submitted so the information was never saved. This made error messages really hard to see as well because they would appear for less than 1 second so I had to screen record and slow down the video to see the errors. Another big challenge I had was changing the HTML of the cart page. I didn't know that getElementByClassName gave an array so I thought I could directly change the inner HTML of the element I obtained. But that didn't work because I was trying to add HTML to a whole array, instead of the actual element which I could get by indexing into the array.

5 lessons I learned

1. localStorage
    a. Before this project, I didn't know how to save information across pages because once I refresh a page, the newly added information would just disappear. But with localStorage, I'm able to store this locally and use it later on. After learning about this, I also went to other shopping websites and saw how they used localStorage. In this assignment, I used localStorage for storing the number of items in cart and the items in the cart. You can see this by clicking inspect > application > localStorage.
2. Data structure
    a. Initially, I thought about hardcoding the drink properties since I only had one drink page working. But I quickly realized that with the cart page, it would be much better to implement classes for these drink objects. This also made me think about what the necessary attributes of this class are. For instance, in addition to the basic properties like quantity and flavor, I added a total price attribute that is the product of quantity and price. This will make the displaying cart total on the cart page much easier. Another attribute I added later on was "tag". Since there are many images for the ingredients and flavors, I had to make a tag for each unique drink in order to display the images without hardcoding.

3. How to debug / implement complicated features
    a. Many times throughout this project, I got bugs after implementing new features and sometimes these new features impacted the previous working ones so I didn't know how to fix / what caused the errors. This made debugging very difficult and I had to rollback to the previous commit a few times. After making the mistake of completing several steps at once when making a new feature, I learned to do it step by step — break the feature down into small chunks and test after each chunk is added. This way, it's much easier to debug and see where the error is.
4. Where to call javascript file
    a. Prior to this assignment, I've always called the javascript file in the head (where I call the stylesheet). Because of this, I had the issue where the javascript will run before the content is loaded. I was stuck for a while because I didn't know why I couldn't get any elements of a class when they are clearly there. After doing some research, I found that I should include the javascript file at the end of the body section of the HTML file. This way, all the javascript functions will only run after the HTML content is processed.
5. How to add event listeners
    a. When I added new items to the cart in script.js file, I wasn't sure how to call on javascript functions with parameters. I needed access to localStorage for some of the functions which I didn't have. To fix this, instead of directly calling the function in the HTML, I looped through the elements with a particular class name, and added event listeners to them. This way, I could pass in the parameters I needed. An example of this is the delete button and the plus/minus buttons on the cart page.