PUI Reflection 6B

**Reflection:**

While working on this assignment I came across several, that can be avoided during the next assignment. The bugs can be divided into two types of errors: syntax errors and logic errors.

**Logic Errors:**

While creating the dynamic cart page in JavaScript, I used my HTML cart page as a reference to create the various variables. However, a major mistake, I made while creating these variables was naming multiple variables with the same name. In my HTML file, I used classes with the same name for various parts to make styling components easy in CSS. As I was trying to create a new HTML element and attach it to the DOM tree, I assigned it to variables with the same name as the class name to remember them easily. However, I forgot that variables have to be unique to avoid writing over them. This became a major problem because I had multiple variables with the same name. I went through my code again and changed each variable name to be unique. For example, while I was creating the variable to create a new HTML element for the organization of the glaze and quantity buttons, I named them “cartButtonsOrganization”. When I edited the code, I named them “cartButtonsOrganizationGlaze” and “cartButtonsOrganizationQuant” respectively.

Furthermore, I named many of my classes and ids very similarly. I had to constantly go back through my code to figure out what named I used when creating a new HTML element through document.createElement in the function loadProducts(). It was very tedious; had I spent more time upfront naming I wouldn’t have spent so much time understanding where I used each name. In the future, I can avoid going back and renaming by naming things specifically to their intended function and ensuring that each name is unique.

Next, I made several mistakes with the append.Child function. I consistently mixed up the order of the placement of the parent and child in the function. While the parent should come first and then the child within the parentheses, I switched the order. For example, the “minusButton” was a child of the parent “cartButtonsOrganizationQuant”, however I organized the function as minusButton.appendChild(cartButtonsOrganizationQuant), rather than cartButtonsOrganizationQuant.appendChild(minusButton). Additionally, I organized the functions in the incorrect order, concerning separating the ‘div’ block, so the items were rendered in the wrong order. I could mitigate this error in the future by commenting on what to do in advance thoroughly.

**Syntax Errors:**

While setting up the local storage I added an unnecessary ‘s’ to the key, this caused an issue when trying to save and push data to the local storage. I named the key “productsitem” instead of the intended “productitems”. When I pushed the Bun object into the local storage, it produced an error. Furthermore, when I tried to get the items from the local storage and assign to the selected array it returned an error that “productitems” did not exist. I had to correct the code to enable items to be retrieved from and push into the local storage.

**Programming Concepts**

Five programming concepts that I learned in JavaScript and used in this assignment were:

**JSON:**

I learned that JSON is used to exchange data between a browser and a server. Understanding JSON allowed me to develop a better conceptual understanding of how to create and store data and how data formatting worked. I specifically used JSON.parse and JSON.stringify to convert text into a JS object and convert a JS object into string respectively. In conjunction with the local storage, I used this to store and retrieve objects. When I added items to my local storage, I used JSON.stringify to convert my selected object into a string. This is needed because local storage can only save items in strings and I had objects and numbers.

**Local Storage:**

I learned how to use local storage to store values of items that were clicked on the product detail page to be reflected in the cart page dynamically. I used getItem to access the items in the local storage and setItems to push items into the local storage. I used local storage to store things like the number of times add to cart was clicked, the quantity of cinnamon rolls that were selected, the glaze of the cinnamon rolls, the subtotal, and the total price.

**Data Types:**

I learned how to use various data types to load the cart page dynamically. For example, I used document.createElement (‘data type’) to create a new HTML element in JS and match it to the data type in HTML. For example, when constructing a button I reflected the data type to be a button, document.createElement(‘button’), then used .setAttribute and matched it with the class name from my HTML code to make sure that the CSS still ran correctly.

**Objects:**

The most valuable piece of information that I learned in this assignment was learning how to use and create objects. I used objects to store all the values of each selection on the product detail page to save each selection a user made. I used the object ‘Selected’ on my product detail page that was used to update the details of the selection on the cart page. I used my object to fill each of my innerHTML to update the content to match the selection made by the user.

**Commenting:**

I commented by JS code to understand what steps to take to ensure that I carried everything out correctly. I didn’t realize how efficient this was until a friend suggested using this method. I began by commenting on all of the steps I would need to accomplish a set function and then carried that out with actual code. I think it’s helpful when tracking back to understand where changes need to be implemented.