Assignment 6B Reflection

Andy Wilbourn

GitHub Link Website

Challenges

I faced a few different challenges along the way here. The most daunting to me was dynamically generating new divs or containers of content within the cart page. I was able to copy the divs I had from my hard-coded cart page from assignment 6A, and that actually worked out nicely. The key for me was figuring out how to nest the divs within one another. This process required the use of a single line of code within an innerHTML call on the javascript for the page loading. This way, the code generated would nest properly and be influenced by the CSS styles I had associated with each of the classes on the prior page.

Another challenge I faced was getting the subtotal, tax cost, and total cost values to update after I removed an object from the cart. The page would regenerate without the container of the removed object, but the values for the costs wouldn't update. So I ended up having to utilize a page refresh in order to get the values to where they need to be.

Programming Concepts

1. innerHTML: I learned about generating code for within a specific code block in a page's HTML using dynamic creation via innerHTML within javascript. This method allowed for the generation of my basket page's contents, and allowed for the removal function to work as well, as we have to regenerate the remaining content dynamically.

2. localStorage: I utilized localStorage as the method of storing information from the product detail page about what type of item people were purchasing, and then pushing that information across different pages on my site. In this case, I pulled information in by parsing an array of objects from local storage onto my basket page, after stringify-ing that same array on the product detail page.

3. Objects: products that people were interested in purchasing went into an object, in this case containing details about the product's size, color, quantity to purchase, and total cost of that quantity. This information was needed in order to generate a shopping basket page with the actual items users select for purchase. I generated the objects in JS using functions that automatically updated values as users selected different parts of the product detail.

4. Alerts: I utilized a few different alerts this time, including one that asks the user to confirm the they are removing an item from the cart. This consequential action felt like it needed a specific confirm() so I generated one using an if-else statement.

5. for() statements: I utilized for statements as a way to loop through my array of objects and generate HTML for each object within that array. It was necessary because that array won't always be the same size when the user is moving into the basket page, so making the code adapt is necessary.