Diana Zhan Shum

Assignment 6B

Web Prototype with JavaScript

GitHub link: dianazhans.github.io/assgn6/

In my prototype the user is able to do the following:

- View their shopping cart page by clicking on the shopping cart icon in the top right corner of the header
- See the items they've added to their cart with information that's reflective of their selections
- Remove any items from the cart followed by an accurate update of the subtotal and number of items in the cart

Reflection

During this assignment I encountered two major challenges: (1) retrieving and using a locally stored array, and (2) creating copies of an HTML element.

The first challenge I faced was correctly retrieving the locally stored array of "products". This was an array I used to save all the items the user decided to add to their cart. During my initial try I was able to access the array, however, it was returning as a string. For example, when I requested to access the first item of the array (i.e. index 0), it would return a bracket " [". Turns out I was accessing the first component of the string instead of the actual item. To better understand what was going on, I looked at the type of the object and learned it was an "HTML collection". I then googled how to access an "HTML collection". With this research I came to understand that I was accessing a string, which was not the correct form. I referred back to the lab slides and recalled the concept of "parse". I implemented that line of code, ensuring the "products" array was parsed prior to using it. This solved the issue and I was finally able to access the first item (i.e. object) along with its properties. This challenge really highlighted the importance of ensuring that when retrieving arrays from local storage, they should be in the correct format to be used. This will definitely be helpful in the future as it's likely that other website experiences will need to carry data from one page to another, meaning retrieval from local storage will be common. I can now more confidently tackle that use case.

The second challenge I faced was creating copies of an HTML element — particularly cloning the same visual representation for each item in the cart. In the design, each item is represented the same way when accessing the shopping cart. The only things that are different are the properties for each item. For example, the flavor of the roll may be different depending on what the user has decided to add to their cart. They may have a cinnamon roll flavored Pumpkin Spice and another item flavored Blackberry. Initially, I tried to use to clone() method, but was unsuccessful. I then googled alternative ways of creating copies of an element and replicating the content inside. I was able to find some guidance on Stack Overflow. I ended up testing the concept and implementing it in the assignment. The process was as follows: (1) create a div element, (2) assign properties, such as a class in order to ensure the same CSS styling is applied, (3) copy the innerHTML of the original div and replicate it in the new div (4) append this new div as a child to the body of the shopping page. This process worked with my website and I was able to successfully create copies of the HTML element and use them as a template for visually representing the items in the cart.

This assignment has been the most challenging so far, however, looking at it retrospectively I have realized that I've learned so much. Most importantly, I was able to learn all these concepts using a realistic and common scenario — the online shopping experience.

Programming Concepts

1. **Arrays**
   I used an array to collect the items the user has decided to add to their cart. This array, called "products", is locally stored and then used in the shopping cart to display the user's selections. Using an array also enable me to loop through the items that were stored to accurately show the items users have selected.

2. **Stringify**
   I used local storage to save the array of "products" and in order to do this I used the concept of "stringify". This was something taught during lab and I was able to apply it in this assignment. Stringify converts the JavaScript array and into a JSON string. I then saved this to local storage so I could access it in the shopping cart page.

3. **Parse**
   In order to retrieve and use the array stored in local storage, I applied the concept of “parse”. This was also something taught during lab. Parse allowed me to retrieve data from the web server (i.e. “products” array that’s locally stored) in the form of a string and convert it to a JavaScript object. I was then able to run a for loop through the array in order to load and populate the items in the shopping cart.

4. **Constructors, Prototypes and Objects**
   To create and add the selected items to the “products” array I used a constructor to create a prototype — a template that would help create the various objects of the items. I assigned the prototype with properties that were common across the items, such as: flavor of roll, glaze, image and price. The content for each property would be different for each item, but using a prototype allowed me to keep the structure for each item the same.

5. **For Loop**
   In the shopping cart page, I used a for loop to access the different items stored in the “products” array in order to run additional code that would help populate them upon page load. Over the summer, we completed a course on Python and covered for loops. I found this to be helpful as the conceptual part had been previously covered. However, the syntax was different and I was able to learn the difference during this assignment.