Jocelyn Gao
PUI
Reflection
March 25, 2020

1. Write a one (or more) paragraph reflection (due with Assignment 6B)
    - What challenges or bugs did you encounter?
    - How did you overcome these challenges?

    While trying to implement Javascript into my website, I faced many challenges since I have never used Javascript prior to this class. The first challenge I faced was in 6A when I was trying to update the product description page based on the selections that the user made. I struggled finding a way to immediately change the product's total cost on the page based on the quantity that the user chose in the dropdown menus. I knew that I needed to use an event handler as the foundation of the code that would recognize when the user selected/changed their selection for the quantity of buns they wanted, but I didn't quite know what code/keyword to use for this function. To overcome this challenge, I did a lot of research online on event listeners and event handlers in Javascript, and came across the "onchange" handler, which I ended up using for my code.

    The second challenge I encountered was finding a way to add items to the shopping cart that accurately reflected what the user selected. First, I tried to use local storage, keys, and parsing to save what users had selected when they pressed the "add to cart" button, but I struggled majorly with finding ways to store information and more so on how to transfer the data between pages. To overcome this challenge, I decided to use modals that open over the menu page for the product description "page" and shopping cart "page" to negate the need for local storage.

    The final major challenge I faced occurred in 6B when I was trying to add the capability of removing an item in the shopping cart. First, I figured out how to remove the row in the table by using "event.target" and ".parentNode". However, this would only temporarily display the accurate shopping cart items the user wanted because my code did nothing to the "shoppingCart" array that held all of the shopping cart items, and the shopping cart display function is based on this array. I then struggled with finding a way to get the row number of the clicked "x" button since I wanted to use the row number to get the index of the item that should be removed from the array. To overcome this challenge, I did a lot of research online about how to get the row number of a clicked button and looked over sample codes to try to understand it and apply it to my code. Once I figured out how to get the row number, I was able to remove that selected item from the "shoppingCart" array with the ".splice" method to accurately display items in the user's shopping cart and the cart's subtotal.

2. What programming concepts did you learn as a part of the assignment?
    - Illustrate at least 5 concepts with an example.
    1. Objects: I created an object for each “order” the user submits that contains the customizations of the order that the user selected (customizations include quantity and glazing), and that required me to gain a thorough understanding of objects and how to save data to them. In Javascript, I first created an empty object saved as the “singleProduct” variable, and then pushed information to the object (the product’s name, glazing, quantity, price, and a button for removing items) based on the user’s selections by using “singleProduct.*attribute*” for each type of information.
    2. Arrays: After creating an object for each order, I saved each added-to-cart order in a “shoppingCart” array by first creating an empty array at the start of each session, and pushing each object ordered to this array with the “.pop” method in order to save them. I then used this array and the information it contained to accurately display the items and details of each order that the user added to their cart. Additionally, in order to access certain objects in the array (for removing items), I had to understand the structure and relationships within the array (such as index), and used the “.splice” method to remove items from the shopping cart.
    3. “for” loops: For my “displayShoppingCart()” function, I used the “for” loop to maintain consistency in how the information in the “shoppingCart” array would be displayed for each order in the table on the shopping cart “page.” Specifically in my code, for each order in the “shoppingCart” array, I created a row in the table used to display the items, created a cell for each attribute of the object, and updated each empty cell with the customizations the user selected, all in the same order for consistency (facilitated by using the “for” loop). This also allowed me to use only one function to call on each order in the “shoppingCart” array regardless of how many updates are made to the array.
    4. Scope: For some functions in my code, I used local scope when defining variables, and for others I used global scope. For example, I used global scope for the “shoppingCart” array since I knew that this is a variable that I would call upon in more than one function. In contrast, I defined some variables locally like “singleProduct” since I would only use this variable for the “AddtoCart()” function, and only wanted to be able to access the variable with this function.
    5. DOM Node Properties: In order to accurately update the user’s shopping cart based on the item(s) they remove, I needed to understand the structure of the table and node properties like “.parentNode” and node methods like “.removeChild”. Specifically in my code, I used “event.target.parentNode” to get the column that the clicked “remove” button is in, then used “.parentNode” again to get the row that the clicked button is in, and another “.parentNode’ to get the object (entire row of data) that the clicked “remove” button correlates to. Then I used the “.removeChild” method to remove the entire row of data from the shopping cart page.