

Karen Wu Section D

Website Link: https://karenwu8742.github.io/homework_6b/

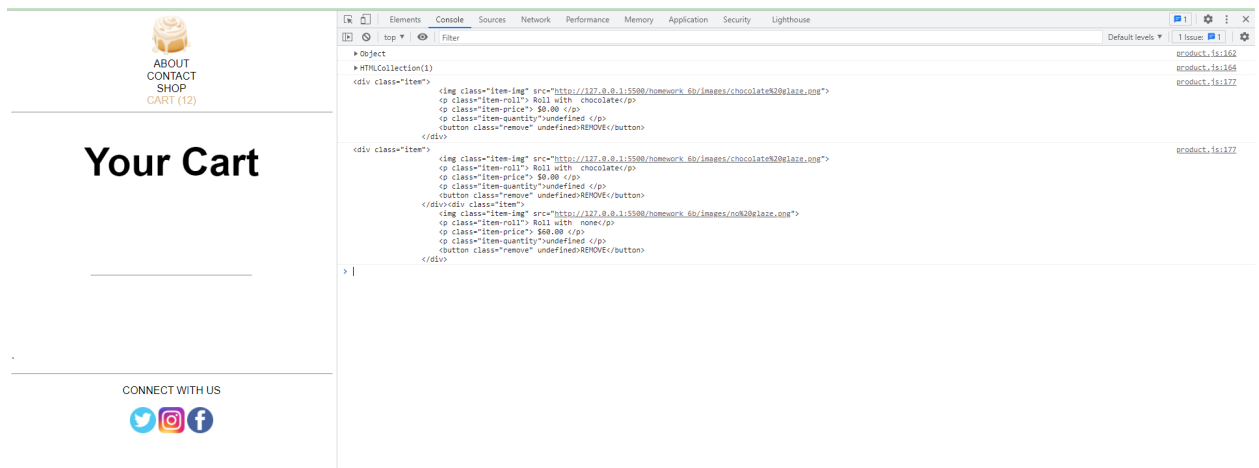
Source Code: <https://github.com/karenwu8742/karenwu8742.github.io>

Reflection

For this assignment, I had a lot of problems with displaying the cart to the point that I couldn't resolve the issue. I spent hours debugging, but was still not sure why my code was not displaying in the cart page even though I had the HTML code correctly implemented with no bugs. I wrote up the code for removing items from the cart, but since my items won't display, I could not debug that. I think for the future I will definitely try to understand javascript more since I feel that it is definitely a small piece of code that I am missing or have written wrong and if that was changed, I would definitely be able to finish the project. I also had trouble with localStorage and spent a lot of time debugging for that to work first. I was able to get that to start working by researching and understanding how it works. At first, I was unable to see whether my items were being added so I incorporated numbers on the page which lets me see how many items are in the cart and localStorage. I learned about using the application in inspect to delete the storage so I could continue debugging without items displayed in my cart. My code runs without error, but I think there can be a lot that can be improved on.

Programming Concepts

1. A programming concept that I learned was using inspect and console to debug. I continually used "console.log()" throughout my code to make sure things were running correctly and the correct functions were called on. Also by looking at the console, I was able to see where my errors were and how I could potentially fix them, but it was a lot harder when there was no error. By looking at inspect, I got a better grasp of javascript.



2. Another programming concept is adding event listeners in my Javascript and HTML code. I was able to use this when clicking buttons and it was widely used throughout my code to trigger events.

```
<h1>Original Roll</h1>
<div class="product">
  <div class="details">
    <div class="description">
      
    </div>
    <div class="description">
      <h3>Glaze</h3>
      <button class="glaze" onclick="sugarMilkGlaze()">Sugar Milk</button>
      <button class="glaze" onclick="vanillaGlaze()">Vanilla Milk</button>
      <button class="glaze" onclick="chocoGlaze()">Double Chocolate</button>
      <button class="glaze" onclick="noGlaze()">None</button>
    </div>
  </div>
</div>
```

3. One programming concept that I learned was being able to retrieve and manipulate pre-existing HTML. I used this to update the numbers for how many items were in the cart and adding items. I also used it to update prices, but for things that did not display it was a lot harder to work with.

```
// Update the cart on the navigation bar
function updateCart() {
  let cartNumStored = JSON.parse(localStorage.getItem('cartNum'));
  console.log("Hello")
  let selectedQty = parseInt(document.getElementById('qtyNum').value);
  let cartNum = document.getElementById('cart').text;
  cartNum = cartNum.replace('CART (', '');
  cartNum = cartNum.replace(')', '');
  if (cartNumStored) {
    localStorage.setItem('cartNum', JSON.stringify('CART (' + (selectedQty + parseInt(cartNum)) + ')'));
    document.getElementById('cart').text = 'CART (' + (selectedQty + parseInt(cartNum)) + ')';
  }
  else {
    localStorage.setItem('cartNum', JSON.stringify('CART (' + selectedQty + ')'));
    document.getElementById('cart').text = 'CART (' + selectedQty + ')';
  }
  displayCart()
}
```

4. The next programming concept is accounting for edge cases. There were many times where different variables or retrieves resulted in NULL and this really makes it hard to not have an error as you cannot do anything to NULL variables. Since this happened many times, I learned to always account for edge cases. I did this by writing if statements to whether or not the variable was NULL so a different case would occur if it were. It really helped decrease the number of errors I had in my code.

```
// Display the number of rolls in the cart
function cart() {
    let cartNumStored = JSON.parse(localStorage.getItem('cartNum'));
    if (cartNumStored) {
        let currCart = document.getElementById('cart');
        currCart.textContent = (cartNumStored);
    }
    let priceSum = JSON.parse(localStorage.getItem('priceSum'))
    if (priceSum === null) {
        localStorage.setItem('priceSum', JSON.stringify(0))
    }
}
```

5. The last programming concept is connecting different functions. There are times where I needed to have function calls within a function and this helped with having things happen asynchronously. For example, when I want to update what options to customize for the roll, the price should change at the same time or after an item is added to the cart, the number of items should change.

```
// Update the price on product page
function updatePrice() {
    let currPrice = document.getElementById('currPrice');
    let qty = document.getElementById('qtyNum').value;
    let price = (5) * parseInt(qty);
    currPrice.innerHTML = 'Total: $' + parseFloat(price);
    localStorage.setItem('currPrice', JSON.stringify(parseFloat(price)))
    updateTotal()
}
```