Yang Cheng
PUI Section E
Assignment 6b

Link: https://yangcheng-design.github.io/homework_6b/

Extra Credit:

Related products carousel on the bottom of the product details page

Errors/How I Resolved Them

Using Document Methods

In the beginning of the project, I was getting familiar with the Document methods such as getElementsbyClassName, querySelector, I got the syntax wrong. For example, document.querySelector takes in ".className" similar to a CSS selector while document.getElementbyId takes in the name of the ID.

I resolved these bugs by printing out the HTML element every time I used one of these methods and reading the MDN documentation to make sure I knew what the method returned (array versus single element). In the future, I would write programs in smaller chunks and practice using methods in a smaller-scoped environment such as a CodePen, to get familiar with them in advance.

Managing Different JS Files

I used three .js files: main.js managed the cart icon number while single-product.js and cart.js managed the corresponding pages. I got some errors when I wasn't sure which function was updating the cart icon number. The cart number is updated when user adds to cart and removes from cart.

I solved this issue by printing out using console.log in every function that touched the cart icon. I was eventually able to trace the problem. In the future, I would design my program in advance so that I had a better idea where variables are referenced and updated.

Climbing the DOM Tree

I ran into a few errors while using multiple ".parentElement" to get higher in the DOM tree. My code selected the wrong element at times.

I solved this issue by printing out via console.log, and by inspecting in the Firebox Developer Console. The Developer Console generates the dynamically rendered HTML elements, which was very helpful to trace the DOM tree.

Five Programming Concepts

Local Storage
I used local storage for the first time in this project. For my website, I used local storage to keep every product user added to cart (and their color and size properties). Then on the Cart page, I retrieved the items in local storage and iterated through the array to display the products on the page. The structure of local storage took some getting used to, but it proved to be a very powerful element for web programming.

Local vs. Global Variables
I learned to better distinguish JS variables const, let, var in this project. I ran into errors trying to reassign a const and then read up on the differences online. I used let whenever using an iterator in a for loop. I used const and var when I had a variable I didn't want to modify, for example the array of Product objects.

Event Handlers
I used event handlers to run a function when a button is clicked. For example, when the "Add to Cart" button is clicked, the event handler stored the item in local storage then incremented the number of items in cart. I also used a for loop on the color and size buttons, iterating through them and listening for a click. When a click is detected, it triggered a function that added the "active" class to the element which changed its appearance.

CSS Animations
I used CSS animations for the first time on the Add to Cart button. When the user hovers over the button, it would invert the border and background colors in a slow transition. This animation signals that this button is clickable and adds delight to the interface.

The Switch Statement
To pull the correct product image for each color in the shopping cart, I used a switch statements with four cases, one for each color, i.e.

```
switch(colorString) {
        case "strawberry":
            return './images/backpack-strawberry.png';
        case "blackberry":
            return './images/backpack-blackberry.png';
        case "crazyberry":
            return './images/backpack-crazyberry.png';
        case "orange":
            return './images/backpack-orange.png';
}
```

Switching on cases was much more effective than multiple if statements in my case.