David Domalik PUI Spring 2021 4/8/2021

Adding Functionality with JavaScript

Assignment 6

Reflection

Having used JavaScript before, I didn't have many major challenges but did run into several bugs. One was that I was using jQuery's toggleClass function when hovering on the menu page, but if the mouse was already on the page when it loaded, the toggle was reversed. I realized that I didn't need JavaScript for this feature and replaced it with CSS's built-in hover selector to solve this. Doing math also presented some challenges, for example calculating the price of an item as different options were selected on the details page. My early iterations didn't respond if a customization was removed after being added, so knowing when to subtract from the price became an issue. Unfortunately my solution involved hard-coding some default quantities into a switch statement. Math was also an issue on the checkout page; when I summed up the price of all the items in the cart, I saw strange results. I realized the error was that JavaScript was actually performing string concatenation instead of float addition, which was hard to detect since variables are dynamically typed. After casting to the correct type, the math resolved itself.

Programming Concepts

1. This project allowed me to refine my file structure for creating a website. For example, in this project, I created folders to separate the images, CSS, and JS from the HTML pages. I also split my CSS and JS into multiple files to make things more manageable and avoid one incredibly long file that would be hard to navigate. The separated files each correspond to a different HTML file and only the necessary files are imported for that page.
2. It was good to revisit commenting so that I had clean code. I learned that one should both create sections within the code (examples in common.css) as well as explain complex lines that are hard to understand on first glance (examples in the loadTable function in checkout.js). Good comments made my code much easier to navigate, especially as I worked on the project at various times over the past few weeks.
3. I put special care into making sure I abstracted functions wherever possible. This greatly helped me reduce copy & paste errors and made it easy to change code when I encountered an issue. The JS functions updatePrice, calculatePrice, and createTableRow are all good examples of where I abstracted code that is called in multiple places throughout the file.
4. I had never used CSS grid before this class, but it came in handy for formatting my header. I've traditionally used Flexbox but that was difficult to work with in this case because I wanted to leave the left side of the header empty and keep the title/logo centered in the page. This required a three-column grid where each column was equal width, and I could control the positioning of the elements inside each column. The CSS grid was perfect for the job.
5. I was also new to using localStorage. I used localStorage to store the items a customer added to their cart and needed data to stay consistent from page to page. The localStorage cart can be found throughout my JavaScript code.