

## Reflection

1. When I tried to add items to the cart, I didn't know how to change the position and content based on storage using javascript. I created an element in html and appended new text node to the element in javascript. However, nothing showed up when I run the code. I then realized I have to create new elements in the loop, append text to that new element, and then append the new element to the original parent. Thus, the text nodes can exist individually with different position and content instead of being squeezed into one parent. Figuring out this bug helped me better understand the definition of parent and child, and I'll know when to append elements in the future.
2. When I tried to show total number of items in cart, the number was wrong. I stored the amount of items added in local storage after each "Add to cart" action. However, I found that the numbers were not added together correctly even though I had the right equation. For example, 1+2 became 12. I realized it might be the case that the numbers were added as strings. I found that the "number" I parsed out of storage were actually strings after storing them with JSON.stringify. Thus I used Number() to convert them to number before doing calculation. For future programming work, I would remember to convert local storage to correct type after parsing to avoid this kind of mistakes.
3. When I tried to delete item from product list, no matter which delete button I clicked on, I could only delete the last item. I found that the index I used to delete item was always the index of the last item. It was because the index was saved when the loop ended. I solved this problem by saving the index as a property of the element during the loop, so that the index won't change as loop changes. In later assignments, I will know how to keep track of the index and store it when needed.

## Programming concepts:

1. onLoad: In class we learned how to use onLoad() to call functions when we load the page. For this assignment, I used onLoad in every javascript file to call the functions I want to run. Furthermore, to do the delete function, I called onLoad inside a function to immediately update the page.
2. addEventListener: We learned the concept of event listener and how to use this function to interact with the web page. For this assignment, I used addEventListener multiple times to implement interactions such as "Add to cart" and "delete". My addEventListener function takes in "click" and another self defined function, which helped me define what the action of clicking does to the webpage.
3. localStorage: We learned how to store the information we need in local storage. This concept becomes very useful to the implementation of cart and wish list. When user clicks on "Add to cart" and "Add to wish list", I firstly used "parse" to get the array from local storage, and then I updated the array and store it back to local storage. For the cart and wish list page, I just needed to get the array I stored and loop through them to display.
4. appendChild: we also learned how to append child to a parent in class. For the cart wish list page, I created multiple new elements and appended them to a parent during the loop. I used this concept to display the images and text node of each item added to cart and wish list. For each individual child, I can define their individual position and style.
5. Object: In class we learned how to write functions for objects and define different attributes for objects. For this assignment, I put four different pillows into four objects and assigned them attributes of "name", "color", "picture", and "quantity". I could store all these information together with object in array. When I needed to display a certain object, I could also have access to the information associated with it.