Issues I encountered

1) One bug I had was not being able to switch between different sub-categories in the product detail page. After clicking one of the sub-categories (e.g. big size), the category kept being selected. After referring to stack overflow and youtube videos, I came up with the solution that using 'flag' to tie the sub-categories to selectors and disabled the selected style of the unselected category while enabling the style of the selected one.

2) Another critical issue I encountered was not being able to remove items from the cart. After referring to series of youtube videos, I solved the problem by using ".remove" when the item number <= 0.

3) I also had difficulty adding unit "USD" after the total money in the cart page. I tried editing the html file first to add the string "USD", but it didn't work. Finally, I identified the reason was that I set the function of counting totoal money in js file, and js has a higher priority than html. The only way to add the unit "USD" was to edit the function in my js file.

Programming Concepts I learned

1) Flag: used as a signal to let the program know that a certain condition has met. In this example, I used flag = "big"/"small"to represent the selection of different sub-categories.

```
selectors[0].onclick = function () {
    flag = "big"
    DetailImage.src="Detailbig.png"
    for (var j=0;j<selectors.length;j++){
        selectors[j].style.borderColor=''
        selectors[j].style.color=''
        selectors[j].style.fontWeight=''
    }
    this.style.borderColor='#923D00'
    this.style.fontWeight='bold'
}
```

2) localStorage.setItem & localStorage.getItem: these two methods allow you to store and retrieve values in the localStorage object. I used these concepts to count and store the value of product numbers.

```
carts[0].onclick = function () {
    if (flag == "big") {
        if(bigs) {
            localStorage.setItem('Bigs', bigs + 1)
            bigs = parseInt(localStorage.getItem('Bigs'));
            span.innerHTML=bigs+smalls
        }
        else {
            localStorage.setItem('Bigs', 1);
            bigs = parseInt(localStorage.getItem('Bigs'));
            span.innerHTML=bigs+smalls
        }
    }
```

3) parseInt: converting its first argument to a string, parses that string, then returns an integer or NaN

4) addEventListener: the addEventListener() listen for an event and can execute a callback function when a button is clicked, as the example below shows.

```
for (let i=0; i<selectors.length; i++){
  selectors[i].addEventListener('click', () => {
    selectPrice(products[i]);
  })
}
```

5) document.querySelector: a method that returns the first HTML element that matches the specified selectors.

```
function selectPrice(){
  let productNumbers = localStorage.getItem('cartNumbers');
  productNumbers = parseInt(productNumbers);

  carts.forEach(add => {
    add.addEventListener('click', () => {

    if(productNumbers) {
    localStorage.setItem('cartNumbers', productNumbers + 1);
    document.querySelector('#cart span').textContent = productNumbers + 1;
  } else {
    localStorage.setItem('cartNumbers', 1);
    document.querySelector('#cart span').textContent = 1;
    }

    });
  });
}
```