The biggest challenge I encountered was ensuring my cart number stayed updated even if I moved to different pages. I first tried to replace my navigation bar HTML with JavaScript so that I could display the same code in the navbar.js file across my website. Below is the code I initially wrote.

```javascript
navbar = '<div id="home"><a class="homenav" href="index.html">Home</a></div>'+
'<ul class="ulnav">' +
    '<li class="linav"><a class="productnav" id="spacing" href="productlist.html">Products</a></li>' +
    '<li class="linav"><a class="aboutnav" id="spacing" href="#">About</a></li>' +
    '<li class="linav"><a class="cartnav" href="cart.html">Cart</a></li>' +
'</ul>'

$('#header').html(navbar);
```

However, this did not solve the problem. I realized that a more appropriate solution was to use the localStorage property (I referenced this Youtube video). I used this same method multiple times in my code to make updates and save the update to be reflected on HTML. I made comments on my JavaScript file so I could copy and paste the code and just change the references and targets.

```javascript
arrayOfItems = JSON.parse(localStorage.getItem("globalCart")); //get global
arrayOfItems.push(newOriginal); //add new item
localStorage.setItem("globalCart", JSON.stringify(arrayOfItems)); //update global
```

```javascript
$('#addtocart').click(function(){
    var count = localStorage.getItem("cartNumber"); //get global
    count++; //update count
    localStorage.setItem("cartNumber",count); //update global
    updateCartIndicator();
```

```javascript
function minusCart(){
    var cart = localStorage.getItem("cartNumber"); //get global
    cart--; //update cart
    localStorage.setItem("cartNumber",cart); //update global
    updateCartIndicator();
}
```

What I learned from this was that if I am able to understand one concept well, I am able to duplicate its usage for multiple purposes with just slight changes in the code. When I was first assigned this homework, there seemed to be so much logic I needed to figure out, but it turned out that I could solve multiple tasks with similar codes.