1. What challenges or bugs did you encounter? How did you overcome these challenges?
   I start this assignment with relatively few knowledges in JavaScript except for what TA had taught us in the lab. There were a lot of challenges in transferring my logical understanding of the code into actual code in line. The biggest challenge among these is figuring out the way to display the stored items in the shopping cart. I had a hard time figuring out how to display multiple objects in the array in order and delete each item card according to the button user clicks. I was originally only displaying one item in the array and linking item property into Html page section one by one. In this way, I was only recalling and displaying the last item from the local storage, and when I remove the object from the array, the Html code will remain in the place with the undefined value. Then I search in google and realize the correct way to call multiple items in local storage is by looping through the array and display them one by one. To display each item in the same format without repeating the HTML code multiple times, I found a simpler way by using "template literals" to quote HTML code directly in JavaScript file. In this way, I was able to generate each item card one by one in inside the loop, without repeating the same HTML code multiple times. I found the loop method a very useful strategy in doing repeated work, together with array and storage, this method could be used in the future in many places where the task is repetitive and with a large amount.

2. What programming concepts did you learn as a part of the assignment?
   a. **The LocalStorage concepts**:
      it's a type of web storage that enables JavaScript to store and access data locally in the browser. The stored data could be retrieved from another local webpage locally. I used the LocalStorage to store an array of pillows which their properties and retrieve those pillow properties and display them in the shopping cart page, using localStorage.setItem(key name, value); and JSON.parse(localStorage.getItem("name")

```
10: null
▶ 11: {name: "Travel Pillow", price: "$54", quantity: …
▶ 12: {name: "Travel Pillow", price: "$54", quantity: …
▶ 13: {name: "Travel Pillow", price: "$54", quantity: …
▼ 14: pillow
    color: "Morning Haze"
    fill: "Hypoallergenic Poly-blend"
    image: "pink.jpg"
    name: "Travel Pillow"
    price: "$54"
    quantity: "3"
  ▶ __proto__: Object
  length: 15
▶ __proto__: Array(0)
>
```

   b. **template literals**:
      it is a string literal that allows me to embed HTML expression in JavaScript. I used it to embed my HTML code that displays the cart items properties into the JavaScript so the same HTML code doesn't need to be repeated. I used ${array[i].value} to insert the item property values

into the HTML code to make them shown properly on a web page.

```
cartItems += `<div class="item" id="itemOne">
            <input type="checkbox" name="additem" value="added"><br>
            <img src="${cartItem[i].image}" alt="pillow" id="itemImage" style="width: 100%">

            <div class="description">
               <span id="itemName">${cartItem[i].name}</span>
               <span id="itemColor">${cartItem[i].color}</span>
               <span id="itemFill">${cartItem[i].fill}</span>
            </div>

            <div class="quantity">
               <button class="plus-btn" type="button" name="button" onclick="addQuant()">+</button>
               <input type="text" name="name" id="itemQuant"value="${cartItem[i].quantity}">
               <button class="minus-btn" type="button" name="button" onclick="minusQuant()">-</button>
            </div>

            <div class="price" id="itemPrice">${cartItem[i].price}</div>
               <button type="submit" class="buttonDelete" id="delete" href="#" onclick="deleteItem()">delete</button>
               <button type="submit" class="button" href="#">add to wishlist</button>
            </div>`;
}
```

c. change html display content inside JavaScript—*element.innerHTML=""*

it is another API what works in HTML or XML that containing element property. I used the *document.getElementById("storedItem").innerHTML= cartItems[i]* to firstly locate my target section with id"storedItem" and insert the new value"cartItem[i]" into the target html section. In this way the JavaScript could modify the value and display content.

d. JavaScript Object Constructor:

It is the function in JavaScript that initialize an object. I used it to initialize an object to store my pillow properties such as name, price and other information. When the condition for generating a new pillow is met, using *new pillow (property1, property2, property3, property4) with the same property order shows up in the constructor, a new object will be generated.*

```
function pillow(name, price, quantity, color, fill, image){
    this.name = name;
    this.price = price;
    this.quantity = quantity;
    this.color = color;
    this.fill = fill;
    this.image = "pink.jpg";
}
```

e. removing object from array using JavaScript Array splice () method.

this is the JavaScript inbuilt method that allow us to modify content in array. I used this method to delete the objects in my shopping cart array, and delete the whole item in my web page using code: *document.getElementById("storedItem").innerHTML=cartItem.splice(i, 1);*