

Interactive Shopping Cart 🛒

Anisa Jibrell

Assignment 6B | PUI Fall 2021

Bug #1

Bun Bun	Quantity	Subtotal
 Bun Bun  {item.name} ← Glazing: Vanilla Milk Price: \$4.00 Delete Item	6	\$24.00
 Bun Bun {item.name} Glazing: None Price: \$3.00 Delete Item	1	\$3.00

Many of the sources of my issues took place in the `getCart()` function where I used the backtick to embed JavaScript variables into a string.

One bug that occurred was when the name of the cinnamon bun wouldn't show up in the cart. I realized that it was because the dollar sign in front of `{item.name}` was missing as shown below.

Bug #2

Another issue I encountered was when the cart items weren't being pushed to the cart page as shown below. I eventually realized it was because I forgot to call the `getCart()` function on the `cart.html` page.

Your Cart		
Bun Bun	Quantity	Subtotal
?	?	?
		Subtotal
		Tax
		Total

```
<script type="text/javascript" src="...>
</head>

<body onload="getCart()">

<!--
-->

<div class="product-backdrop">
  <div class="container">
```

Concept #1

1. Dynamically generate content

One of the core concepts I learned as part of this assignment was how to **dynamically generate content**.

Prior to this assignment, all of my cinnamon products were situated on separate pages - which made my code **more complicated** than it needed to be.

Through numerous lessons with my tutor I gained tips that I was able to carry over to my assignment, such as building an inventory through creating **an array of objects** in order to dynamically generate a menu.

This allowed me to have one detail page, versus one for every cinnamon bun.

```
1
2
3 //creating an array of objects for the inventory to dynamically generate a menu. creating a unique id for each item
4 const inventory = [
5   {id:1, name: "Original", price:3.00, img:"https://i.ibb.co/dbf4vsT/Cinnamon-bun-new.png", //original
6     description:"Handcrafted with sweet dough and cinnamon-sugar filling.",
7     glazing: ["None", "Sugar Milk", "Vanilla Milk", "Double chocolate"]}, //glazing property has four possible values
8   {id:2, name: "Blackberry", price:4.00, img:"https://i.ibb.co/RQd7vNH/blackberry.png",
9     description:"Handcrafted with sweet dough and cinnamon-blackberry filling.",
10    glazing: ["None", "Sugar Milk", "Vanilla Milk", "Double chocolate"]},
11   {id:3, name: "Walnut", price:4.00, img:"https://i.ibb.co/gWdfytb/walnut.png",
12     description:"Handcrafted with sweet dough and cinnamon-walnut filling.",
13     glazing: ["None", "Sugar Milk", "Vanilla Milk", "Double chocolate"]},
14   {id:4, name: "Original (Gluten-Free)", price:4.00, img:"https://i.ibb.co/dbf4vsT/Cinnamon-bun-new.png", //original
15     description:"Handcrafted with gluten-free dough and cinnamon-sugar filling.",
16     glazing: ["None", "Sugar Milk", "Vanilla Milk", "Double chocolate"]},
17   {id:5, name: "Pumpkin Spice", price:4.00, img:"https://i.ibb.co/02PYnYw/pumpkin-spice1.jpg",
18     description:"Handcrafted with sweet dough and pumpkin-sugar filling.",
19     glazing: ["None", "Sugar Milk", "Vanilla Milk", "Double chocolate"]},
20   {id:6, name: "Caramel Pecan", price:3.00, img:"https://i.ibb.co/QKQWwDn/carmel-pecan1.jpg",
21     description:"Handcrafted with sweet dough and caramel pecan filling.",
22     glazing: ["None", "Sugar Milk", "Vanilla Milk", "Double chocolate"]},
23 ];
24
25 ];
```

There are four possible values assigned to the glazing property.

Concept #2

1. Retrieving products individual ID

Through this assignment, I learned how to extract the individual id of a product from a query string.

```
// Query String - Bansan Workshop, details.html?id=1  
const id = parseInt((window.location.href).split("id=")[1]);  
// id = 1
```

In the example above, within the `getDetails()` function, I split a query string based on the token `"id="`. Essentially what this means, is everything to the left of `"id="` is index 0 and everything to the right is index 1.

```
const item = getItemById(id)
```

This then allows me to retrieve out of the inventory array based on id.

Concept #3

1. Template literals

The use of template literals, which are enclosed by the backtick(`) allowed me to embed variables into a string.

When building rows in the cart, this made it easier to add more content to the row.

```
let row = `${trStart}<div><p>${item.name}</p><p>Glazing: $
```

Concept #4

1. Delete item from cart

One of the tricky parts of the assignments was figuring out how to delete an item from the existing cart.

I achieved this by creating an empty cart and looping throught the existing cart with a for loop. When user clicks "Delete Item" it checks if the index of that item is equal to index of the item to delete.

When building rows dynamically, javascript:deleteFromCart calls the deleteFromCart() function when 'Delete Item' is clicked on the selected item in the gv_cart array.

Then all items except the one to delete are added to the new cart. It then replace current cart with new cart that has all items except the deleted item.

```
        }
        function deleteFromCart(indexToDelete)
        {
            let newCart=[] // create a new empty array
            for (let i in GV_CART) //loop through the array
            {
                if (i != indexToDelete)//when user click delete
                {
```

Concept #5

1. Local Storage

Adding and retrieving objects from local storage, and understanding that local memory has to be stored as a string.

```
var GV_CART=[]; //empty array
var GV_ITEM={}; //declaring as an empty object
function loadCart(){ // load cart from local storage
    const shoppingCart = localStorage.getItem("ShoppingCart"); //getting data
    // alert(shoppingCart);
    if (shoppingCart) // if there is a shopping cart in local storage.
    {
        | GV_CART = JSON.parse(shoppingCart); // convert JSON String to array
        // alert(shoppingCart);
    }
    else { // if there's nothing found in local storage
        | return 0; //then return 0
    }
    document.getElementById("cart-items").innerHTML = GV_CART.length;
    return GV_CART.length;
}
```

References

<https://stackoverflow.com/questions/27678052/usage-of-the-backtick-character-in-javascript>

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Template_literals

<https://stackoverflow.com/questions/1085801/get-selected-value-in-dropdown-list-using-javascript?rq=1>

<https://developer.mozilla.org/en-US/docs/Web/API/Window/localStorage>

Extra Credit: Product pages for all your products, including all feedback and functionality