Assignment 6 – Use Web Storage for Cart

Lab A & B due: WEDNESDAY, October 26, 2022. 11:59 pm ET
Lab C & D due: MONDAY, October 24, 2022. 11:59 pm ET
This is an **individual** assignment.

Note: The assignment length takes the midterm quiz and the fall break into account. We **recommend that Labs C & D should finish this assignment before the fall break**, as the next assignment (Final Project proposal) will be presented in the Wednesday labs after the fall break.

1 Learning objectives

- Further advance your JavaScript skills
- Practice simple data storage using JavaScript local storage

2 Tasks

In this last assignment for the Bun Bun bake shop, you will make the shopping cart data persist. Similar to previous homeworks, you can use the following Roll class declaration.

```
class Roll {
    constructor(rollType, rollGlazing, packSize, rollPrice) {
        this.type = rollType;
        this.glazing =  rollGlazing;
        this.size = packSize;
        this.basePrice = rollPrice;
    }
}
```

For the rest of this assignment, you will edit the existing script files from HW4 and HW5 to implement the behaviors below.

2.1 Product detail page

When the page loads, attempt to retrieve the cart from the local storage. If no cart exists in the storage, create an empty `cart` array.

When the user has made their glazing and pack size selection and clicked on "Add to Cart," perform the following actions:

(1) Similar to HW4, create a Roll instance containing all of the current product information and add it to the `cart` array.
(2) Convert the updated cart to JSON, save it in the local storage, and print the current contents of the cart in local storage after saving.

2.2 Cart page

When the cart page is loaded, perform the following actions:

(1) Attempt to retrieve the cart from the local storage. If no cart exists in the storage, create an empty `cart` array.
(2) Populate the DOM with all of the items in the current cart, using your code from HW5.

When the user clicks on the “Remove” link at any item,

(1) Remove the item from the `cart` array.
(2) Convert the updated cart to JSON, save it in the local storage, and print the cart in local storage after saving.
(3) Similar to HW5, remove the corresponding entry from the DOM and update the total price field at the bottom of the page.

Additional Notes:

(1) The cart may contain identical items (i.e., two rolls with the same type, glazing and pack size). Make sure that when you click on the “Remove” link at one item, only that particular item is removed.
(2) Check that the cart in the local storage persists as you reload or navigate to different pages.
(3) Make sure the cart in the local storage is always up-to-date. As per the above instructions, you should print out the cart in local storage whenever it is updated. During grading, we will look at the printed cart data to check the correctness of your code.

2.3 Use proper code style

Follow the code style instructions from previous assignments, i.e., external `js` files, no external libraries, follow code style guidelines.

3 Submission

(1) **Deploy** your webpage on GitHub.
Check that your code is online (**pushed**) & that your webpage works properly online.

(2) Submit your homework repository to **Gradescope**.
Verify that it was submitted to Gradescope before the deadline!