

# Assignment 5 – Add & Delete items from Cart

Lab A & B due: Tuesday, October 11, 2022. 11:59 pm ET

Lab C & D due: Thursday, October 06, 2022. 11:59 pm ET

This is an **individual** assignment.

## 1 Learning objectives

- Further practice your JavaScript skills. Practice loops and manipulating objects.
- Practice adapting your webpage dynamically based on user input: adapt and show updates.

## 2 Tasks

In this assignment, you will add some basic functionality to the Bake Shop **shopping cart** page.

The user will start with four items in their cart. You will use JavaScript to **display** those four items on the shopping cart page. When the user clicks one of the “remove” buttons, the corresponding item should be **removed** from the page. The total price, shown at the bottom of the page, should be **updated**.

### 2.1 Initialize the shopping cart

- (1) Create either an [array](#) or a [set](#) to represent your cart.
- (2) Create a `Roll` class (or copy it from HW4). Your class should look like this:

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

- (3) Using the `Roll` class, make four new `Roll` objects and add them to your cart. The four cart items should be:

	Type	Glazing	Pack Size	Calculated Price
a.	Original	Sugar Milk	1	\$2.49
b.	Walnut	Vanilla Milk	12	\$39.90
c.	Raisin	Sugar Milk	3	\$8.97
d.	Apple	Original	3	\$10.47

To clarify, the `Roll` class `basePrice` property holds the *base price* for one single roll, similar to HW4. The `Roll` **objects** that you create here based on this table should calculate the item price using the roll base price, glazing price and pack size. You should re-use the price computation code from HW4 here. *Do not hard-code the calculated price.*

## 2.2 Display cart items on the page

- (4) Write a function that takes a `Roll` as an argument, and appends the appropriate DOM elements to the shopping cart page (see [mockup](#) from HW2). You may find [<template>](#) elements to be useful here, though you are not required to use them. For each cart item, be sure to include:
  - (4.1) The correct picture for the roll.
  - (4.2) The name of the item (Original, Raisin, Walnut, etc.).
  - (4.3) The glazing for the item (Original, Sugar Milk, etc.).
  - (4.4) The pack size (1, 3, 6, 12).
  - (4.5) The item price you calculate (not the roll base price).
  - (4.6) A “remove” button for the item.
- (5) For each item in your `cart`, call your function from Step 4 and display the items on the shopping cart page. Make sure to update the total price field as well.

## 2.3 Remove items from the cart

- (6) When users click the “Remove” link at any entry,
  - (6.1) the item should be deleted it from the `cart` array or set,
  - (6.2) removed from the DOM,
  - (6.3) and the price total at the bottom should be updated.
- (7) Make sure to test your implementation against edge cases, e.g., when your cart is empty (`cart.length == 0`).

## 2.4 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!