

COMSATS UNIVERSITY ISLAMABAD, ATTOCK CAMPUS

JAVASCRIPT SHOPPING CART SYSTEM

SUBMITTED BY: IRAM SAJJAD

SP22-BSE-023

SUBMITTED TO: SIR KAMRAN

ASSIGNMENT NO: 01

DATE: 26-09-2024

Introduction:

Implement a shopping cart in JavaScript using ES6 arrow functions, array methods (map, filter, reduce), and object manipulation. The cart is capable of adding, removing, updating items, calculating the total cost, and applying discounts using arrow functions.

Code Explanation:

1. Add Items to Cart:

This function allows us to add products to the cart. It accepts parameters like product ID, product name, quantity, and price, and adds the product to an array representing the cart.

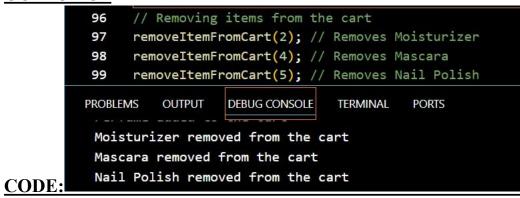
OUTPUT OF

```
//checking working of code
          87
          88
                // Adding products to the cart
                addItemToCart(1, 'Lipstick', 1, $255);
          89
                addItemToCart(2, 'Moisturizer', 1, $30);
           90
                addItemToCart(3, 'Foundation', 1, $400);
          91
                addItemToCart(4, 'Mascara', 1, $4020);
          92
                addItemToCart(5, 'Nail Polish', 2, $4015);
          93
                addItemToCart(6, 'Eyeshadow Palette', 1, $4035);
          94
                addItemToCart(7, 'Perfume', 1, 50);
                             DEBUG CONSOLE
                                           TERMINAL
          PROBLEMS
                    OUTPUT
                                                     PORTS
           C:\Program Files\nodejs\node.exe .\MAD.js
           Lipstick added to the cart
           Moisturizer added to the cart
           Foundation added to the cart
           Mascara added to the cart
           Nail Polish added to the cart
           Eyeshadow Palette added to the cart
           Perfume added to the cart
CODE:
```

2. Remove Item from Cart:

This function removes an item based on the product ID. It uses array methods like findIndex and splice to remove the product.

OUTPUT OF



3. Update Item Quantity:

This function updates the quantity of a specific product by searching for its product ID and using map to modify the quantity.

OUTPUT OF CODE:

```
// Updating quantities of products in the cart
100
      updateItemQuantity(1, 5); // Update Lipstick to 5 units
101
102
      updateItemQuantity(3, 2); // Update Foundation to 2 units
      updateItemQuantity(6, 3); // Update Eyeshadow Palette to 3 units
103
      updateItemQuantity(7, 1); // Update Perfume to 1 unit
PROBLEMS
          OUTPUT
                   DEBUG CONSOLE
                                  TERMINAL
                                             PORTS
                                                             Filter (e.g. text, !exclude, \escap
 Quantity updated for product with ID 1
 Quantity updated for product with ID 3
 Quantity updated for product with ID 6
 Quantity updated for product with ID 7
```

4. Calculate Total Cost:

This function uses the reduce method to sum up the total cost of all products, considering their quantity.

5. Display Cart Summary:

This function lists all the items in the cart, showing the product name, quantity, and total price.

6. Apply Discount:

An optional feature that applies a discount code to the total price.

OUTPUT OF CODE:

```
Calculating total cost
106
      calculateTotalCost();
107
      /**Displaying the cart summary*/
      displayCartSummary();/**Applying a discount*/
108
      applyDiscount('DISCOUNT10');
109
          OUTPUT
                   DEBUG CONSOLE
                                  TERMINAL
PROBLEMS
                                            PORTS
                                                            Filter (e.g. text, !ex
 Quantity updated for product with ID 6
 Quantity updated for product with ID 7
 Total Cost: 360
 Cart Summary:
 Product: Lipstick, Quantity: 5, Total Price: 125
 Product: Foundation, Quantity: 2, Total Price: 80
 Product: Eyeshadow Palette, Quantity: 3, Total Price: 105
 Product: Perfume, Quantity: 1, Total Price: 50
 Total Cost: 360
 Total after applying discount: 324
```

Conclusion:

This assignment taught me how to effectively handle a shopping cart using JavaScript functions such as map, filter, and reduce. I also got hands-on experience with arrow functions and applying discounts. One of the challenges I encountered was updating the cart and making sure the total price was calculated accurately, which really helped me enhance my debugging skills.