Assignment Title:

JavaScript-based Mobile Shopping Cart

Name : Qadir Nawaz Khan

Reg No:Sp22-Bse-005

Date:25 September 2024

INTRODUCTION :

Objective of the Assignment

The goal of this assignment is to create a simple shopping cart system using JavaScript. It lets users add, update, remove items from the cart, and calculate the total cost of all items. The program uses modern JavaScript features like arrow functions and array methods to perform these tasks.

Operations Implemented

1. **Add Items**: A function to add products (with details like name, quantity, and price) to the cart.
2. **Remove Items**: A function to remove items from the cart by their product ID.
3. **Update Quantity**: A function to update the quantity of products in the cart.
4. **Calculate Total**: A function to calculate the total cost of all items in the cart.
5. **Display Cart Summary**: A feature that shows the list of items in the cart with their quantity and total price.

Code Explanation:

- addItemToCart Function:

  - **Purpose**: This function adds a new product to the shopping cart.
  - **How it Works**: It takes product details like productId, productName, quantity, and price as inputs, creates an object to represent the product, and adds it to the cart using the push method.
  - **Why it's Important**: It allows users to add products to their shopping cart.

- removeItemFromCart Function:

  - **Purpose**: This function removes an item from the cart based on the product's ID.
  - **How it Works**: It searches the cart to find the index of the product using its productId. If the product is found, it removes it from the cart using the splice method.
  - **Why it's Important**: It helps users remove unwanted items from their cart.

- updateItemQuantity Function:

  - **Purpose**: This function updates the quantity of an item already in the cart.
  - **How it Works**: It uses the map method to go through each product in the cart. If it finds a matching productId, it updates the quantity of that product.
  - **Why it's Important**: It allows users to change the number of units they want to buy for a specific product.

- calculateTotalCost Function:

  - · **Purpose**: This function calculates the total cost of all the items in the cart.
  - **How it Works**: It uses the reduce method to sum up the cost of each product, considering both the price and the quantity.
  - **Why it's Important**: It provides users with the total amount they need to pay for all the products in their cart.

- displayCartSummary Function:

  - **Purpose**: This function shows a summary of the cart's contents.
  - **How it Works**: It uses the map method to loop through each item in the cart and display its name, quantity, and total price.
  - **Why it's Important**: It gives users a clear overview of what's in their cart and the total price of each product.

- filterZeroQuantityItems Function:

  - **Purpose**: This function removes any items from the cart that have a quantity of zero.

- **How it Works**: It uses the filter method to create a new list of products where the quantity is greater than zero.
- **Why it's Important**: It keeps the cart clean by removing products that are no longer wanted (zero quantity).

- applyDiscount **Function** (Optional):

  - **Purpose**: This function applies a discount to the total price based on a discount code.
  - **How it Works**: It checks if the entered discount code is valid and then applies a discount (e.g., 10% or 20%) to the total cost.
  - **Why it's Important**: It allows users to save money by applying discounts to their total bill.

Screenshots of our code:

```
Laptop added to the cart.
Mouse added to the cart.
Keyboard added to the cart.
Cart Summary:
Laptop (x1) - $1000
Mouse (x2) - $50
Keyboard (x1) - $50
Total Cost: $1100
Quantity of Mouse updated to 3.
Keyboard removed from the cart.
Cart Summary:
Laptop (x1) - $1000
Mouse (x3) - $75
Items with zero quantity removed.
Total Cost with Discount: $967.5
```

CONCLUTION:

Through this assignment, I learned how to create a basic shopping cart system using JavaScript. I gained a better understanding of how to use modern JavaScript features like arrow functions and array methods (map, filter, reduce). These methods made it

easier to manipulate the data in the cart, such as adding, removing, updating items, and calculating the total cost.

The main challenge I faced was understanding how to use these array methods effectively. It took some practice to get comfortable with how map, reduce, and filter work, but after experimenting with them, I became more confident in using them for managing data in the cart.

This assignment helped me improve my JavaScript skills, especially in working with arrays and objects.

Thanks