

# SHOPPING CART IMPLEMENTATION

27-september-2024

ASHAN KHAN sp22-bse-016

## **Introduction**

Objective of the Assignment:

The objective of this assignment is to build a simple shopping cart system using JavaScript. The cart allows users to add, remove, and update items, view the cart summary, calculate the total cost, and apply discounts based on a provided discount code.

Operations Implemented:

- Add Item to Cart: Adds a product to the cart and logs the action.
- Remove Item from Cart: Removes a product from the cart by its product ID and logs the removal.
- Update Item Quantity: Updates the quantity of a specific product in the cart based on its product ID.
- Display Cart Summary: Shows the cart summary, including product name, quantity, and total cost, using `console.table()` for a well-formatted output.
- Calculate Total Cost: Computes the total cost of all items in the cart.
- Apply Discount: Applies a discount based on a discount code and recalculates the total cost after applying the discount.

## **Logic Behind Each Function**

addItemToCart(product):

Purpose: Adds a product to the cart.

Logic: The product object (with properties like id, name, quantity, and price) is pushed into the cart array. A log statement confirms the addition by displaying the product's name.

Removeitemfromcart(product)

Purpose: Removes a product from the cart by its unique id.

Logic: The function uses `findIndex()` to locate the product in the cart array. If found, it uses `splice()` to remove the product from the array and logs its removal. If not found, a message indicating the product wasn't found is logged.

Updateitemquantity(productid,newquantity)

Purpose: Updates the quantity of a specific product in the cart.

Logic: The function searches for the product by id using `find()`. If found, the quantity property of that product is updated to the newQuantity. A log confirms the update, otherwise, it logs that the product was not found.

Dispaycartsummary()

Purpose: Displays the summary of products in the cart.

Logic: The function first filters out products with a quantity of 0 (if any). Then, it uses `map()` to return an array of objects that contain the name, quantity, and total (price \* quantity) for each product. The summary is printed using `console.table()` for a readable tabular format.

Calculatetotalcost()

Purpose: Calculates the total cost of all products in the cart.

Logic: The function uses `reduce()` to iterate through the cart array and accumulates the total by summing the product of price and quantity for each item. The total is logged and returned.

`Applydiscount(discountcode)`

Purpose: Applies a discount based on the discount code provided.

Logic: The function calculates the total cost using `calculateTotalCost()`. A switch statement checks for a valid discount code (DISCOUNT10, DISCOUNT20, or DISCOUNT30), each applying a percentage discount. If the code is valid, the function calculates the discounted total and logs it. If the code is invalid, it logs an error message and returns the original total cost.

Summary of Logic:

Each function manipulates or accesses the cart array, which stores the products. The functions perform specific actions like adding, removing, updating, or summarizing products. Calculations are made based on the quantity and price of the products in the cart, and discounts are applied if valid discount codes are provided.

## Screenshot :

```
C:\Program Files\nodejs\node.exe .\ahsan.js
Laptop has been added to the cart.
Mouse has been added to the cart.
> (2) [{...}, {...}]



| (index) | name     | quantity | total |
|---------|----------|----------|-------|
| 0       | 'Laptop' | 1        | 1000  |
| 1       | 'Mouse'  | 2        | 100   |



Quantity for Mouse has been updated to 3.
> (2) [{...}, {...}]



| (index) | name     | quantity | total |
|---------|----------|----------|-------|
| 0       | 'Laptop' | 1        | 1000  |
| 1       | 'Mouse'  | 3        | 150   |



Laptop has been removed from the cart.
> (1) [{...}]



| (index) | name    | quantity | total |
|---------|---------|----------|-------|
| 0       | 'Mouse' | 3        | 150   |



Total cost: $150.00
Discount applied: 20%. Total after discount: $120.00
```

## Conclusion

Through this assignment, I learned how to manipulate arrays and objects in JavaScript, specifically using functions like `map()`, `filter()`, and `reduce()`. I also improved my understanding of conditional logic with switch statements for applying discounts and handling different scenarios. The main challenge was ensuring correct item updates in the cart, especially when working with object

references and handling cases where products didn't exist. Another challenge was managing total cost calculations and applying discounts without errors. Overall, the assignment strengthened my skills in building dynamic and interactive systems like shopping carts.

