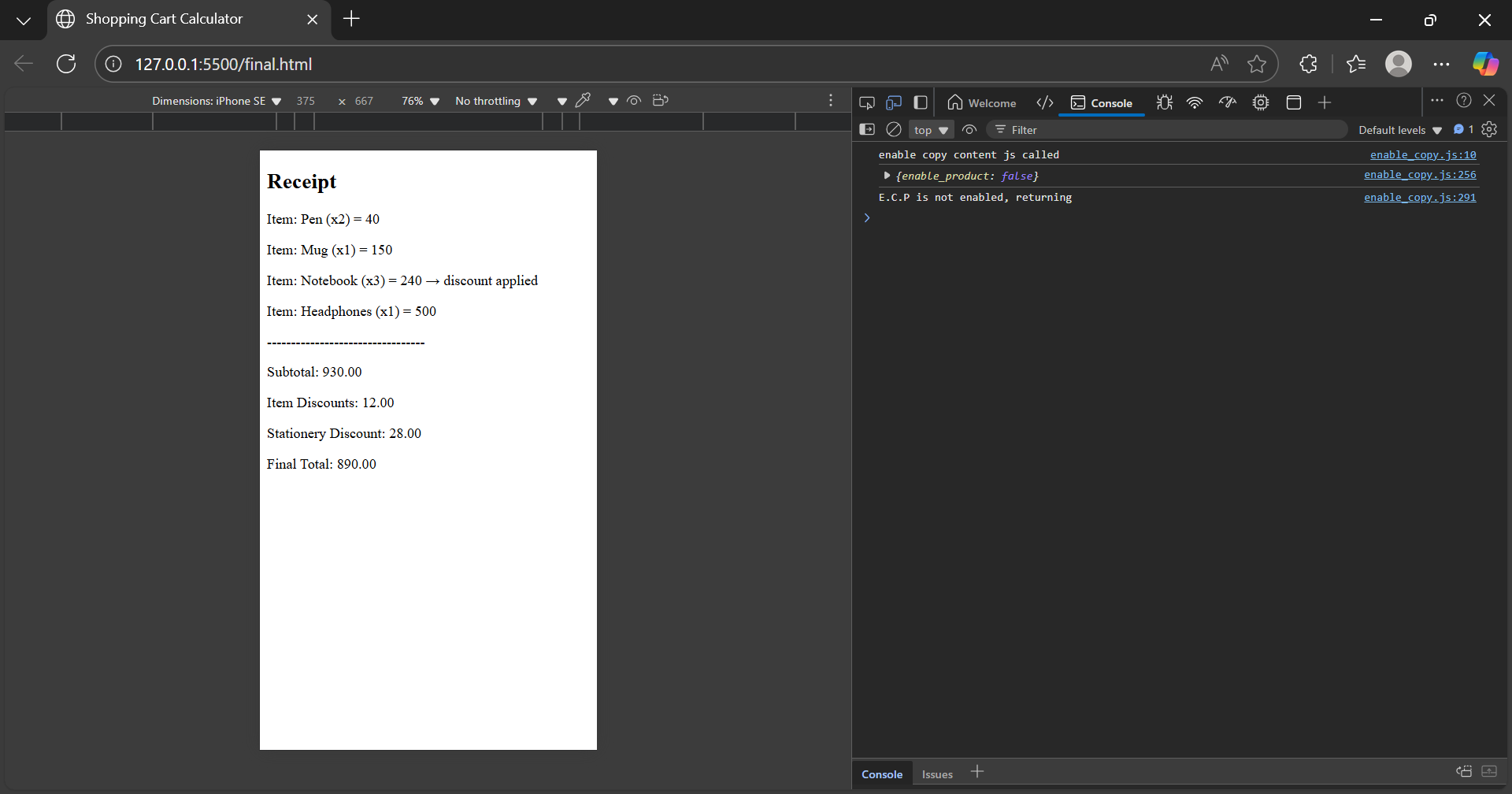
Name : Soumya Dixit

PRN : 23070521151

Batch : B2



<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Shopping Cart Calculator</title>

</head>

<body>

<h2>Receipt</h2>

<div id="receipt"></div>

<script>

// Cart setup

let cart = [

{ id: 1, name: "Pen", price: 20, qty: 2, category: "stationery" },

{ id: 2, name: "Mug", price: 150, qty: 1, category: "kitchen" },

{ id: 3, name: "Notebook", price: 80, qty: 3, category: "stationery" },

{ id: 4, name: "Headphones", price: 500, qty: 1, category: "electronics" }

];

let receiptDiv = document.getElementById("receipt");

// Item-level calculations

let discountedItemTotals = cart.map(item => {

let subtotal = item.price \* item.qty;

if (item.qty >= 3) {

subtotal = subtotal - (subtotal \* 0.05);

}

return subtotal;

});

// Stationery discount

let stationeryItems = cart.filter(item => item.category === "stationery");

let stationeryTotal = stationeryItems.reduce((sum, item) => sum + (item.price \* item.qty), 0);

let stationeryDiscount = stationeryTotal > 200 ? stationeryTotal \* 0.10 : 0;

// Subtotal and discounts

let subtotal = cart.reduce((sum, item) => sum + (item.price \* item.qty), 0);

let itemDiscount = subtotal - discountedItemTotals.reduce((sum, val) => sum + val, 0);

let finalTotal = subtotal - itemDiscount - stationeryDiscount;

// Show receipt

cart.forEach(item => {

let lineTotal = item.price \* item.qty;

let line = document.createElement("p");

line.textContent = `Item: ${item.name} (x${item.qty}) = ${lineTotal}` +

(item.qty >= 3 ? " → discount applied" : "");

receiptDiv.appendChild(line);

});

// Separator

let separator = document.createElement("p");

separator.textContent = "---------------------------------";

receiptDiv.appendChild(separator);

// Totals

let totals = [

`Subtotal: ${subtotal.toFixed(2)}`,

`Item Discounts: ${itemDiscount.toFixed(2)}`,

`Stationery Discount: ${stationeryDiscount.toFixed(2)}`,

`Final Total: ${finalTotal.toFixed(2)}`

];

totals.forEach(text => {

let p = document.createElement("p");

p.textContent = text;

receiptDiv.appendChild(p);

});

</script>

</body>

</html>

**Q1: Which function was hardest to apply? Why?**

Among the functions, reduce() and map() were hardest to apply. reduce() was tricky because it required managing an accumulator for totals, while map() involved applying conditional discount logic instead of simple transformation, making both more complex than filter() or forEach().

**Q2: How could this program be extended?**

The program can be extended by adding tax calculation (e.g., GST), coupon or promo code discounts, and shipping charges based on total cost. Additional features like tiered discounts, loyalty points, or dynamic cart input through user forms would make it more realistic and closer to real-world shopping systems.

let gst = discountedTotal \* 0.18;

let coupon = "SAVE100";

let couponDiscount = (coupon === "SAVE100") ? 100 : 0;

let shipping = discountedTotal > 500 ? 0 : 50;

let loyaltyPoints = Math.floor(discountedTotal / 100);

let finalTotal = discountedTotal + gst + shipping - couponDiscount;

let totals = [

`Subtotal: ₹${subtotal.toFixed(2)}`,

`Item Discounts: ₹${itemDiscount.toFixed(2)}`,

`Stationery Discount: ₹${stationeryDiscount.toFixed(2)}`,

`Discounted Total: ₹${discountedTotal.toFixed(2)}`,

`GST (18%): ₹${gst.toFixed(2)}`,

`Coupon Discount: -₹${couponDiscount.toFixed(2)}`,

`Shipping: ₹${shipping.toFixed(2)}`,

`Final Total: ₹${finalTotal.toFixed(2)}`,

`Loyalty Points Earned: ${loyaltyPoints}`

];