



# Full-Stack Web Development

III CSE/AI-ML NBKRIST VIDYANAGAR.....

---

## Problem Statement: Pharmacy Cart Operations

This problem requires you to build a simple web page for a pharmacy's shopping cart. The goal is to perform calculations on a list of medicines and display the results using core JavaScript array methods.

### 1. Core Information Requirements

The system must handle the following information for each medicine product:

- **Product Name:** The name of the medicine (e.g., 'Paracetamol', 'Cough Syrup').
- **Price per Unit:** The cost of a single unit of the medicine.
- **Quantity:** The number of units of that medicine in the cart.

### 2. Functional Requirements

The web page must perform the following actions automatically upon loading:

- **Display Cart:** A table must be generated to display all products in the shopping cart. For each product, the table should show the **Product Name**, **Price per Unit**, **Quantity**, and the **Total for that Product**.
- **Calculate Final Total:** The total amount to be paid for all products in the cart must be calculated and displayed clearly below the table.

### 3. Technical Specifications

- **Data Structure:** The shopping cart data will be a JavaScript array of objects, defined directly in the script.js file. Each object will represent a medicine with the properties specified above (name, price, count).
- **User Interface:** The interface should be built using basic HTML for structure and CSS for styling. No user interaction (like buttons or input fields) is required.
- **Programming Language:** JavaScript is to be used for all application logic. Specifically, you must use:



## Full-Stack Web Development

III CSE/AI-ML NBKRIST VIDYANAGAR.....

- **forEach()** to iterate through the array and render each product row in the HTML table.
- **reduce()** to calculate the final total amount from the array of objects.

### 4. Testing Requirements

The following test cases must be performed to ensure the system functions correctly:

- **Display Cart Test:**
  - **Test Case:** The application loads with the initial list of products.
  - **Expected Result:** A table is displayed showing all products with their name, price, quantity, and the correct calculated total for each product.
- **Total Amount Test:**
  - **Test Case:** The total amount to be paid is calculated.
  - **Expected Result:** A single, correct numerical value representing the sum of all product totals is displayed.

### 5. Learning Objectives

Upon completion of this project, you will be able to:

- **Core Concepts:** Understand how to work with JavaScript arrays of objects.
- **Array Methods:** Effectively use the `forEach()` method to iterate and manipulate data, and the `reduce()` method to perform a cumulative calculation on an array.
- **HTML/CSS/JS Integration:** Integrate HTML for structure, CSS for styling, and JavaScript for dynamic functionality to create a working web application.
- **Problem-Solving:** Break down a problem (displaying a cart and total) into smaller, manageable tasks and implement solutions using appropriate JavaScript methods.