📘 Case Study: Client-Side Product Sorting & Filtering System (HTML, CSS, JS)

🧾 Overview

An e-commerce platform wants a lightweight product browsing system that allows users to view, sort, and filter product data directly in the browser without relying on any backend or database. This solution should be built purely using **HTML**, **CSS**, and **JavaScript**.

🎯 Objective

To build a browser-based interface that:

- Displays product data from a local JSON-like structure.
- Enables sorting by **price**, **name**, **rating**, **category**, and **stock**.
- Supports filtering by **category** and **rating**.
- Highlights **low stock** items visually for admin monitoring.

📦 Product Data

The product data is stored as a static array of objects in JavaScript:

ID	Name	Price	Rating	Category	Stock
1	Laptop	750	4.5	Electronics	12
2	Phone	500	4.7	Electronics	20
3	Tablet	600	4.3	Electronics	8
4	Monitor	300	4.2	Accessories	15
5	Keyboard	50	4.0	Accessories	30
6	Mouse	25	3.9	Accessories	50
7	Smartwatch	200	4.4	Wearables	10

👥 Users

- **Customer**: Browses and filters products visually.
- **Admin**: Monitors stock levels and top-rated products.

📊 Functional Features

Feature	Description
Product Grid Display	Dynamic rendering of product cards in a grid layout.
Sort by Price/Rating	Dropdown for sorting ascending or descending.
Filter by Category	Buttons or dropdown to filter products by category.
Stock Highlight	Products with low stock (< 10) are styled in red.
Responsive Layout	Grid layout adapts to mobile and desktop views.

📁 Folder Structure

```
product-sort-app/
├── index.html          # UI Layout
├── style.css           # Styling for UI
└── script.js           # JavaScript logic for rendering,
sorting, filtering
```

🖥 UI Layout Plan (HTML)

Section	Description
Header	Title and instructions.
Filter Area	Dropdowns or buttons for sort and filter (category, rating).
Product Grid	Dynamically created cards for each product with name, price, rating, etc.
Footer	Optional credits or messages.

✅ Example Use Cases

1. Customer Sorts by Price (Low to High)

- Customer selects "Price: Low to High"
- JS function reorders the product array and re-renders grid

2. Admin Checks Low Stock Items

- JS marks stock < 10 in red
- Admin can visually spot which items need restocking

3. Customer Filters Electronics

- Dropdown or button selected: `category === 'Electronics'`
- Filtered products shown instantly

💡 Key JavaScript Concepts Used

- DOM Manipulation (`document.createElement`, `appendChild`)
- Event Handling (`addEventListener`)
- Array Methods: `sort()`, `filter()`, `forEach()`
- Template rendering via JavaScript
- Conditional CSS class application (e.g., low stock alert)