🌐 HamiSkills Internship – Web Development Track

Week 3 Task: Modular Shopping Cart System with Persistent Storage for Hami MiniMarket

📖 Project Brief

Hami MiniMarket wants a more professional shopping experience. Your task is to build a **modular shopping cart system** that allows users to add products, manage quantities, and view a dynamic order summary with cart data saved in the browser using **`localStorage`**.

✅ Deliverables

1. **Modular Cart System**

- Create reusable JS modules:
  - `product.js`: handles product rendering
  - `cart.js`: manages cart logic
  - `storage.js`: handles saving/loading from `localStorage`
- Add “Add to Cart” buttons for each product
- Show cart items in a sidebar or modal with:
  - Product name, quantity, price
  - Total cost with tax (e.g., 5%)
  - “Remove” and “Update Quantity” options

2. **Persistent Cart Storage**

- Use `localStorage` to save cart data
- Load cart automatically when the page refreshes
- Show a cart counter in the navbar

3. **Order Summary Page**

- Display a clean summary of selected items
- Include subtotal, tax, and final total
- Add a “Confirm Order” button (no backend yet)

4. **Optional Bonus**

- Add a **toast notification** when items are added
- Animate cart updates (e.g., fade-in, slide-out)
- Add a **discount system** (e.g., 10% off if total > $50)

📂 Submission Guidelines

- Upload your project to **GitHub** with:
  - Clear folder structure (`js/`, `css/`, `assets/`)
  - `README.md` explaining:
  - Features implemented
  - How to test the cart
  - How localStorage is used
- Deploy your site using **GitHub Pages** or **Netlify**
- Submit the live link + GitHub repo on hamiskills.dev under Week 3 tasks

🌍 Professional Practice Requirement

- Post a short **LinkedIn update** with:
  - Screenshot or screen recording of your cart in action
  - Caption explaining your modular code and localStorage usage
  - Submit the LinkedIn post link in the system

🎯 Evaluation Criteria

- Modular code structure (separation of concerns)
- Functionality of cart and order summary
- Use of localStorage for persistence
- UI responsiveness and brand consistency
- Professional presentation (GitHub repo or LinkedIn post)