

Assignment 1 — ShopEase Homepage

Objective:

Build a simple React homepage for an e-commerce app called **ShopEase** to demonstrate component composition, props, and state.

Deadline: Thursday — 11:59 PM (09 /10/ 2025)

Tasks (requirements)

1. **Set up a new React project**
 - Use your preferred tool (Vite, Create React App, etc.).
2. **Header component**
 - Create a Header component that displays the app name (e.g., **ShopEase**).
 - Include a simple navigation bar with at least “Home”, “Products”, and “Cart” buttons (no routing required).
3. **ProductCard component**
 - Create a ProductCard component that accepts the following props:
 - name (string)
 - price (number)
 - image (URL string)
 - The component should display the product image, name, and price using JSX.
4. **Display products on the homepage**
 - Render **3–4** ProductCard components on the homepage with different sample data.
 - You can use placeholder images (Unsplash URLs) or local image files.
5. **Counter component (cart simulation)**
 - Create a Counter component with + and - buttons to simulate adding/removing items from a cart.
 - Each ProductCard should include its own Counter instance to track quantity for that product.
 - Ensure the counter cannot go below 0.

Implementation notes (recommended)

- Use **functional components** and **React hooks** (e.g., useState).
- Keep components small and focused (Header, ProductCard, Counter, App/Homepage).
- Styling: use plain CSS, Tailwind, or any library you prefer. Keep it simple and readable.
- No backend is required — this is a frontend-only assignment.

Deliverables (what to submit)

1. A short README.md in the repo with:
 - Project setup and run instructions (e.g., npm install, npm run dev).

- Any notes about styling or decisions you made.
- 2. Zip File Of Project.

Grading rubric (suggested)

- **Functionality (60%)**
 - Header exists and displays app name (10%)
 - ProductCard uses props correctly (20%)
 - 3–4 ProductCards render on homepage (10%)
 - Counter component works and prevents negative values (20%)
 - **Code quality & structure (25%)**
 - Clear component separation and readable code
 - Proper use of hooks and props
 - **Styling & presentation (15%)**
 - Clean layout and readable UI
 - Proper spacing, alignment, and basic responsiveness
-

Submission instructions

- upload the ZIP file before the deadline.
- Your README.md contains.