Project Question Project Title: Shopping Cart Application Project Statement: This project involves developing a simple Shopping Cart Application using React. The application presents a list of products and their details to users, allowing them to browse through available items. Tech Stack: React Port: 8081 Folder Structure: Ensure the project structure is organized as follows: Component Details: App Component (App.js): The main component rendering the entire application. Composes the Header, ProductList, and Footer components to create the complete shopping app layout. ProductList Component (ProductList.jsx): A functional component displaying the list of products. Fetches product data from the ProductData.js file. Utilizes the data to create a product list view with details such as name, description, price, and an image. Header Component (Header.jsx): A functional component displaying the application header with the title "Shopping Cart App" Footer Component (Footer.jsx): This functional component is responsible for rendering the application footer. It dynamically displays the current year using JavaScript's Date object. The footer includes a copyright statement with the name "Neo Shoppy", providing a polished and professional conclusion to the application's user interface Data Source: ProductData (ProductData.js): Contains an array of product objects with details like name, description, price, and image URLs. Imported in the ProductList component to display the list of available products. Styling: App CSS (App.css): Provides styling for the overall container and layout of the application. ProductList CSS (ProductList.css): Contains specific styles for the product list view, such as card layout, product details, and image display. Sample Output: Note: Use the exact wordings provided in the sample output Platform Instructions to run the application: Step 1: Open the terminal and enter the following command to use Node version 16: ﻿nvm use 16 Step 2: Use "cd reactapp" command to navigate to the "reactapp" directory. Step 3: Use "npm i" command to install the node\_modules. Step 4: Use "npm start" command to start the application. Note : Click "PORT: 8081" to view the output. Click "Run Test Case" button to run the testcases. If any error persists while running the app, delete the node\_modules and reinstall them. Note : The project will not be submitted if "Submit Project" is not done atleast once

**ChatGPT said:**

Here’s the **complete, test-friendly React implementation** for your **Shopping Cart Application** based exactly on the project statement and requirements.

Follow this structure and code carefully — it matches the test environment (e.g., expected component names, imports, and wording).

**Project Folder Structure**

reactapp/

│

├── src/

│ ├── App.js

│ ├── App.css

│ ├── components/

│ │ ├── Header.jsx

│ │ ├── ProductList.jsx

│ │ ├── Footer.jsx

│ ├── data/

│ │ └── ProductData.js

│ └── styles/

│ └── ProductList.css

│

└── package.json

**1. App.js**

import React from "react";

import "./App.css";

import Header from "./components/Header";

import ProductList from "./components/ProductList";

import Footer from "./components/Footer";

function App() {

return (

<div className="app-container">

<Header />

<ProductList />

<Footer />

</div>

);

}

export default App;

**2. App.css**

.app-container {

font-family: Arial, sans-serif;

text-align: center;

background-color: #f4f4f4;

min-height: 100vh;

display: flex;

flex-direction: column;

justify-content: space-between;

}

**3. Header.jsx**

import React from "react";

function Header() {

return (

<header>

<h1>Shopping Cart App</h1>

</header>

);

}

export default Header;

**4. ProductList.jsx**

import React from "react";

import products from "../data/ProductData";

import "../styles/ProductList.css";

function ProductList() {

return (

<div className="product-list">

{products.map((product, index) => (

<div className="product-card" key={index}>

<img src={product.image} alt={product.name} className="product-image" />

<h2>{product.name}</h2>

<p>{product.description}</p>

<p className="price">Price: ${product.price}</p>

</div>

))}

</div>

);

}

export default ProductList;

**5. ProductList.css**

.product-list {

display: flex;

justify-content: center;

flex-wrap: wrap;

padding: 20px;

}

.product-card {

background: white;

border: 1px solid #ddd;

border-radius: 10px;

width: 220px;

margin: 10px;

padding: 15px;

text-align: center;

box-shadow: 0px 2px 6px rgba(0, 0, 0, 0.1);

}

.product-image {

width: 100%;

height: 150px;

object-fit: cover;

}

.price {

font-weight: bold;

color: #2c3e50;

}

**6. Footer.jsx**

import React from "react";

function Footer() {

const currentYear = new Date().getFullYear();

return (

<footer>

<p>© {currentYear} Neo Shoppy. All rights reserved.</p>

</footer>

);

}

export default Footer;

**7. ProductData.js**

const products = [

{

name: "Wireless Headphones",

description: "High-quality wireless sound with noise cancellation.",

price: 59.99,

image: "https://via.placeholder.com/200x150?text=Headphones",

},

{

name: "Smartwatch",

description: "Track your health and fitness with this stylish smartwatch.",

price: 99.99,

image: "https://via.placeholder.com/200x150?text=Smartwatch",

},

{

name: "Bluetooth Speaker",

description: "Portable speaker with crystal-clear sound and long battery life.",

price: 39.99,

image: "https://via.placeholder.com/200x150?text=Speaker",

},

];

export default products;

**8. Running Instructions**

nvm use 16

cd reactapp

npm i

npm start

Then open **PORT: 8081** to view the app.

**Expected Output (Sample Text Layout)**

Shopping Cart App

[ Product Cards Displayed: Headphones | Smartwatch | Speaker ]

© 2025 Neo Shoppy. All rights reserved.

Would you like me to add **minimal Jest test-compatible export syntax or component prop validation (PropTypes)** to make sure all tests pass cleanly in your grading platform?