

Shopping Cart React Project

Project Summary

This project is a **Shopping Cart Web Application** built using **React.js**. It allows users to:

- View a list of products with **images, names, and prices**.
- Add products to the cart and automatically update the **total price**.
- Increase or decrease the **quantity** of items.
- Remove items from the cart.
- See a **live order summary** including subtotal, shipping, and final total.
- Add new products manually by entering **name, price, and image URL** through a form.

The UI design is inspired by **Amazon's shopping cart** with:

- Clean white product cards
- Orange price highlights
- Green "Free Shipping" label
- Yellow "Proceed to Buy" button
- A simple **form to add new items**

This project demonstrates **React state management** (useState), **dynamic rendering with map()**, and **modular component design with CSS styling**.

Steps to Create & Run This Project

1. Install Node.js

Make sure you have **Node.js & npm** installed.

👉 Download Node.js

Check installation:

```
node -v
```

```
npm -v
```

2. Create a React App

In your terminal, run:

```
npx create-react-app shopping-cart
```

```
cd shopping-cart
```

3. Create Components Folder

Inside the src/ folder, create a new folder named **components**.

src/

└─ App.js

└─ index.js

└─ components/

│ └─ ShoppingCart.jsx

│ └─ ShoppingCart.css

4. Add Files

- Copy the **ShoppingCart.jsx** code into src/components/ShoppingCart.jsx
- Copy the **ShoppingCart.css** code into src/components/ShoppingCart.css

5. Update App.js

Replace the code in App.js with:

```
import React from "react";
```

```
import ShoppingCart from "../components/ShoppingCart";
```

```
function App() {  
  return (  
    <div>  
      <ShoppingCart />  
    </div>  
  );  
}
```

```
export default App;
```

6. Run the Project

In your project folder, start the development server:

```
npm start
```

This will open the app in your default browser at:
<http://localhost:3000/>

7. Features Implemented

- ✓ Add, remove, and update item quantities
- ✓ Product images shown in cart
- ✓ User can add **custom products** with name, price & image
- ✓ Real-time total calculation
- ✓ Amazon-inspired UI with CSS styling