// Houseplant Shop Project

// Using React with Redux (Assuming setup is done with Redux Toolkit)

import React from "react";

import { BrowserRouter as Router, Routes, Route, Link } from "react-router-dom";

import { Provider, useDispatch, useSelector } from "react-redux";

import { configureStore, createSlice } from "@reduxjs/toolkit";

// Redux slice for cart

const cartSlice = createSlice({

name: "cart",

initialState: {},

reducers: {

addToCart: (state, action) => {

const id = action.payload;

state[id] = (state[id] || 0) + 1;

},

increment: (state, action) => {

state[action.payload]++;

},

decrement: (state, action) => {

if (state[action.payload] > 1) state[action.payload]--;

else delete state[action.payload];

},

remove: (state, action) => {

delete state[action.payload];

},

clear: () => ({}),

},

});

const { addToCart, increment, decrement, remove } = cartSlice.actions;

const store = configureStore({ reducer: { cart: cartSlice.reducer } });

// Plant data

const plants = [

{ id: "1", name: "Aloe Vera", price: 10, category: "Succulent", image: "🌵" },

{ id: "2", name: "Snake Plant", price: 12, category: "Air Purifier", image: "🌿" },

{ id: "3", name: "Spider Plant", price: 8, category: "Air Purifier", image: "🌱" },

{ id: "4", name: "Peace Lily", price: 15, category: "Flowering", image: "🌼" },

{ id: "5", name: "Cactus", price: 9, category: "Succulent", image: "🌵" },

{ id: "6", name: "ZZ Plant", price: 14, category: "Low Light", image: "🍀" },

];

const categories = [...new Set(plants.map((p) => p.category))];

function Header() {

const cart = useSelector((state) => state.cart);

const itemCount = Object.values(cart).reduce((a, b) => a + b, 0);

return (

<header className="p-4 bg-green-200 flex justify-between">

<h1 className="text-xl font-bold">🌿 Plant Shop</h1>

<nav className="flex gap-4">

<Link to="/">Home</Link>

<Link to="/products">Products</Link>

<Link to="/cart">🛒 {itemCount}</Link>

</nav>

</header>

);

}

function LandingPage() {

return (

<div className="text-center p-10 bg-cover bg-[url('https://via.placeholder.com/800x300')]">

<h2 className="text-3xl font-bold">Welcome to Our Green Haven</h2>

<p className="my-4">We provide beautiful and healthy houseplants to brighten your home.</p>

<Link to="/products" className="bg-green-500 text-white px-4 py-2 rounded">Get Started</Link>

</div>

);

}

function ProductListing() {

const cart = useSelector((state) => state.cart);

const dispatch = useDispatch();

return (

<div className="p-6">

{categories.map((cat) => (

<div key={cat} className="mb-6">

<h3 className="text-xl font-semibold mb-2">{cat}</h3>

<div className="grid grid-cols-2 gap-4">

{plants.filter(p => p.category === cat).map((plant) => (

<div key={plant.id} className="border p-4 rounded">

<div className="text-4xl">{plant.image}</div>

<h4>{plant.name}</h4>

<p>${plant.price}</p>

<button

onClick={() => dispatch(addToCart(plant.id))}

disabled={cart[plant.id]}

className="mt-2 px-2 py-1 bg-green-400 text-white rounded disabled:opacity-50"

>

{cart[plant.id] ? "Added" : "Add to Cart"}

</button>

</div>

))}

</div>

</div>

))}

</div>

);

}

function CartPage() {

const cart = useSelector((state) => state.cart);

const dispatch = useDispatch();

const items = Object.keys(cart).map((id) => {

const plant = plants.find((p) => p.id === id);

return { ...plant, quantity: cart[id] };

});

const total = items.reduce((sum, item) => sum + item.price \* item.quantity, 0);

return (

<div className="p-6">

<h2 className="text-2xl font-bold mb-4">Your Cart</h2>

<p>Total Items: {items.reduce((a, b) => a + b.quantity, 0)}</p>

<p>Total Cost: ${total}</p>

<div className="grid gap-4 mt-4">

{items.map((item) => (

<div key={item.id} className="flex items-center justify-between border p-2">

<div className="flex items-center gap-4">

<div className="text-2xl">{item.image}</div>

<div>

<p>{item.name}</p>

<p>${item.price}</p>

</div>

</div>

<div className="flex items-center gap-2">

<button onClick={() => dispatch(decrement(item.id))} className="px-2 bg-yellow-200">-</button>

<span>{item.quantity}</span>

<button onClick={() => dispatch(increment(item.id))} className="px-2 bg-green-200">+</button>

<button onClick={() => dispatch(remove(item.id))} className="px-2 bg-red-300">🗑️</button>

</div>

</div>

))}

</div>

<div className="mt-6 flex gap-4">

<Link to="/products" className="bg-blue-400 text-white px-4 py-2 rounded">Continue Shopping</Link>

<button className="bg-gray-400 px-4 py-2 rounded text-white">Checkout (Coming Soon)</button>

</div>

</div>

);

}

function App() {

return (

<Provider store={store}>

<Router>

<Header />

<Routes>

<Route path="/" element={<LandingPage />} />

<Route path="/products" element={<ProductListing />} />

<Route path="/cart" element={<CartPage />} />

</Routes>

</Router>

</Provider>

);

}

export default App;