**// context/CartContext.js**

import React, { createContext, useState } from "react";

export const CartContext = createContext();

export const CartProvider = ({ children }) => {

const [cartItems, setCartItems] = useState([]);

const addToCart = (product) => {

setCartItems((prevCart) => {

const existingItem = prevCart.find((item) => item.id === product.id);

if (existingItem) {

return prevCart.map((item) =>

item.id === product.id ? { ...item, quantity: item.quantity + 1 } : item

);

}

return [...prevCart, { ...product, quantity: 1 }];

});

};

const updateQuantity = (id, quantity) => {

setCartItems((prevCart) =>

prevCart.map((item) =>

item.id === id ? { ...item, quantity: Math.max(1, quantity) } : item

)

);

};

const removeFromCart = (id) => {

setCartItems((prevCart) => prevCart.filter((item) => item.id !== id));

};

const clearCart = () => {

setCartItems([]);

};

return (

<CartContext.Provider value={{ cartItems, addToCart, updateQuantity, removeFromCart, clearCart }}>

{children}

</CartContext.Provider>

);

};

**// App.js**

import React from "react";

import { Routes, Route } from "react-router-dom";

import { CartProvider } from "./context/CartContext";

import Navbar from "./Navbar";

import About from "./components/About";

import ProductList from "./components/ProductList";

import ProductDetail from "./components/ProductDetail";

import ShoppingCart from "./components/ShoppingCart";

import OrderConfirmation from "./components/OrderConfirmation";

import "./styles.css";

const App = () => {

return (

<CartProvider>

<div className="container">

<Navbar />

<Routes>

<Route path="/about" element={<About />} />

<Route path="/product-list" element={<ProductList />} />

<Route path="/product/:id" element={<ProductDetail />} />

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

<Route path="/order-confirmation" element={<OrderConfirmation />} />

</Routes>

</div>

</CartProvider>

);

};

export default App;

**// ProductList.js**

import React, { useContext, useEffect, useState } from "react";

import { Link } from "react-router-dom";

import { CartContext } from "../context/CartContext";

const ProductList = () => {

const { addToCart } = useContext(CartContext);

const [products, setProducts] = useState([]);

useEffect(() => {

fetch("https://fakestoreapi.com/products")

.then((res) => res.json())

.then((data) => setProducts(data));

}, []);

return (

<div>

<h2>Product List</h2>

<div>

{products.map((product) => (

<div key={product.id} className="product-item">

<h3>{product.title}</h3>

<p>{product.price} kr</p>

<button onClick={() => addToCart(product)}>Add to Cart</button>

<Link to={`/product/${product.id}`} className="product-detail-link">

View Details

</Link>

</div>

))}

</div>

</div>

);

};

export default ProductList;

**// ShoppingCart.js**

import React, { useContext } from "react";

import { CartContext } from "../context/CartContext";

const ShoppingCart = () => {

const { cartItems, updateQuantity, removeFromCart } = useContext(CartContext);

return (

<div>

<h2>Shopping Cart</h2>

{cartItems.length === 0 ? (

<p>Your cart is empty.</p>

) : (

<ul>

{cartItems.map((item) => (

<li key={item.id}>

<span>{item.title} - ${item.price} x {item.quantity}</span>

<button onClick={() => updateQuantity(item.id, item.quantity + 1)}>+</button>

<button onClick={() => updateQuantity(item.id, item.quantity - 1)}>-</button>

<button onClick={() => removeFromCart(item.id)}>Remove</button>

</li>

))}

</ul>

)}

</div>

);

};

export default ShoppingCart;

**// OrderConfirmation.js**

import React, { useContext, useState } from "react";

import { CartContext } from "../context/CartContext";

const OrderConfirmation = () => {

const { cartItems, clearCart } = useContext(CartContext);

const [orderPlaced, setOrderPlaced] = useState(false);

const handleOrder = () => {

setOrderPlaced(true);

clearCart(); // 주문 후 장바구니 초기화

};

return (

<div>

<h2>Order Confirmation</h2>

{orderPlaced ? (

<p>Thank you for your purchase! Your order has been placed successfully.</p>

) : (

<>

{cartItems.length === 0 ? (

<p>No items in the cart.</p>

) : (

<>

<ul>

{cartItems.map((item) => (

<li key={item.id}>

{item.title} - ${item.price} x {item.quantity}

</li>

))}

</ul>

<button onClick={handleOrder}>Place Order</button>

</>

)}

</>

)}

</div>

);

};

export default OrderConfirmation;