Examen Simulado - ReactJS con useState y useEffect

Este documento contiene una simulación de una aplicación ReactJS con navegación, componentes, formularios y uso de Hooks (useState y useEffect), utilizando rutas modernas.

# 1. App.jsx

import { BrowserRouter as Router, Routes, Route } from 'react-router-dom';  
import Home from './pages/Home';  
import AgregarDestino from './pages/AgregarDestino';  
  
function App() {  
 return (  
 <Router>  
 <Routes>  
 <Route path="/" element={<Home />} />  
 <Route path="/agregar" element={<AgregarDestino />} />  
 </Routes>  
 </Router>  
 );  
}  
  
export default App;

# 2. Home.jsx

import { useEffect, useState } from 'react';  
import DestinoList from '../components/DestinoList';  
import { Link } from 'react-router-dom';  
  
function Home() {  
 const [destinos, setDestinos] = useState([]);  
  
 useEffect(() => {  
 const destinosIniciales = [  
 { nombre: "Machu Picchu", region: "Cusco" },  
 { nombre: "Laguna 69", region: "Áncash" },  
 ];  
 setDestinos(destinosIniciales);  
 }, []);  
  
 return (  
 <div>  
 <h2>Destinos turísticos</h2>  
 <DestinoList destinos={destinos} />  
 <Link to="/agregar">Agregar nuevo destino</Link>  
 </div>  
 );  
}  
  
export default Home;

# 3. AgregarDestino.jsx

import { useState } from 'react';  
import { useNavigate } from 'react-router-dom';  
  
function AgregarDestino() {  
 const [nombre, setNombre] = useState('');  
 const [region, setRegion] = useState('');  
 const navigate = useNavigate();  
  
 const handleSubmit = (e) => {  
 e.preventDefault();  
 if (!nombre || !region) {  
 alert('Todos los campos son obligatorios');  
 return;  
 }  
  
 console.log('Destino agregado:', { nombre, region });  
 navigate('/');  
 };  
  
 return (  
 <form onSubmit={handleSubmit}>  
 <h2>Agregar destino</h2>  
 <input value={nombre} onChange={(e) => setNombre(e.target.value)} placeholder="Nombre del destino" />  
 <input value={region} onChange={(e) => setRegion(e.target.value)} placeholder="Región" />  
 <button type="submit">Guardar</button>  
 </form>  
 );  
}  
  
export default AgregarDestino;

# 4. DestinoList.jsx

function DestinoList({ destinos }) {  
 return (  
 <ul>  
 {destinos.map((destino, i) => (  
 <li key={i}>{destino.nombre} - {destino.region}</li>  
 ))}  
 </ul>  
 );  
}  
  
export default DestinoList;