--- FILENAME: App.test.js  
import { render, screen } from '@testing-library/react';  
import App from './App';  
  
test('renders learn react link', () => {  
 render(<App />);  
 const linkElement = screen.getByText(/learn react/i);  
 expect(linkElement).toBeInTheDocument();  
});  
  
  
--- FILENAME: App.css  
.App {  
 min-height: 100%;  
 display: flex;  
 flex-direction: column;  
}  
  
  
  
.App-logo {  
 height: 40vmin;  
 pointer-events: none;  
}  
  
@media (prefers-reduced-motion: no-preference) {  
 .App-logo {  
 animation: App-logo-spin infinite 20s linear;  
 }  
}  
  
.App-header {  
 background-color: #282c34;  
 min-height: 100vh;  
 display: flex;  
 flex-direction: column;  
 align-items: center;  
 justify-content: center;  
 font-size: calc(10px + 2vmin);  
 color: white;  
}  
  
.App-link {  
 color: #61dafb;  
}  
  
@keyframes App-logo-spin {  
 from {  
 transform: rotate(0deg);  
 }  
 to {  
 transform: rotate(360deg);  
 }  
}  
  
  
--- FILENAME: App.js  
import React from 'react';  
import Header from './components/Header';  
import Body from './components/Body';  
import Footer from './components/Footer';  
  
function App() {  
 return (  
 <div className="App">  
 <Header />  
 <Body />  
 <Footer />  
 </div>  
 );  
}  
  
  
export default App;  
  
  
--- FILENAME: body.css  
.container {  
 margin-top: 100px;  
 margin-left: 15px;  
 margin-right: 15px;  
 display: flex;  
 flex-direction: column;  
 align-items: center;  
 min-height: calc(100vh - 165px);  
 }  
   
 .column {  
 background-color: #F5F5DC;  
 border-radius: 10px;  
 width: 100%;  
 height: 80px;  
 margin: 10px;  
 display: flex;  
 flex-direction: row;  
 justify-content: space-between;  
 align-items: center;  
 padding: 10px;  
 box-sizing: border-box;  
 }  
   
 .image {  
 width: 50px;  
 height: 50px;  
 object-fit: cover;  
 }  
   
 .text {  
 flex-grow: 1;  
 margin-left: 10px;  
 margin-right: 10px;  
 display: flex;  
 flex-direction: column;  
 justify-content: center;  
 }  
   
 .text h2 {  
 margin-bottom: 15px;  
 font-size: 14px;  
 line-height: 1.2;  
 }  
   
 .text p {  
 margin: 0;  
 font-size: 12px;  
 line-height: 1.2;  
 }  
   
 .button {  
 background-color: #FFB6C1;  
 color: white;  
 font-weight: bold;  
 padding: 5px 10px;  
 border: none;  
 border-radius: 5px;  
 height: 100%;  
 width: 100px;  
 box-sizing: border-box;  
 display: flex;  
 justify-content: center;  
 align-items: center;  
 text-align: center;  
 }  
   
 @media screen and (min-width: 769px) {  
 .column {  
 flex-direction: row;  
 height: 300px;  
 }  
   
 .image {  
 width: 200px;  
 height: 80%;  
 }  
   
 .button {  
 width: auto;  
 height: auto;  
 margin-top: 0;  
 }  
 }  
   
  
--- FILENAME: Body.js  
import React, { useState } from 'react';  
import productsData from './db.json';  
import './body.css';  
  
function Body() {  
 const [searchTerm, setSearchTerm] = useState('');  
  
 const handleSearch = event => {  
 setSearchTerm(event.target.value);  
 };  
  
 const filteredProducts = productsData.products.filter(product =>  
 product.title.toLowerCase().includes(searchTerm.toLowerCase())  
 );  
  
 return (  
 <div>  
 <input  
 type="text"  
 placeholder="Pesquisar produtos"  
 value={searchTerm}  
 onChange={handleSearch}  
 className="search-bar"  
 />  
 <div className="container">  
 {filteredProducts.map(product => (  
 <div key={product.id} className="column">  
 <img src={product.image} alt="Produto" className="image" />  
 <div className="text">  
 <h2>{product.title}</h2>  
 <p>{product.description}</p>  
 </div>  
 <button className="button">COMPRE</button>  
 </div>  
 ))}  
 </div>  
 </div>  
 );  
}  
  
export default Body;  
  
  
--- FILENAME: Footer.css  
.footer {  
 background-color: #FFB6C1;  
 height: 165px;  
 color: white;  
 padding: 20px;  
 display: flex;  
 flex-direction: column;  
 align-items: center;  
 justify-content: space-between;  
 }  
  
.footer-content {  
 align-self: flex-start;  
}  
  
.footer-credits {  
 font-size: 15px;  
 text-align: center;  
 font-weight: bold;  
}  
  
@media screen and (max-width: 768px) {  
 .footer {  
 height: 100px;  
 }  
  
 .footer-content {  
 align-self: center;  
 }  
  
 .footer-content h2, .footer-content p {  
 font-size: 12px;  
 margin: 2px 0; /\* Reduzir a margem entre o tÃ­tulo e os parÃ¡grafos \*/  
 }  
  
 .footer-credits {  
 font-size: 10px;  
 margin: 2px 0; /\* Reduzir a margem ao redor dos crÃ©ditos \*/  
 }  
}  
  
@keyframes fadein {  
 from { opacity: 0; }  
 to { opacity: 1; }  
}  
  
  
--- FILENAME: Footer.js  
import React from 'react';  
import './Footer.css';  
  
function Footer() {  
 return (  
 <footer className="footer">  
 <div className="footer-content">  
 <h2>Mundinho Doce</h2>  
 <p>Telefone: (14) 99770-2116 (14) 99814-1078</p>  
 <p>Entregas e Pedidos pelo WhatsApp ;)</p>  
 </div>  
 <p className="footer-credits">Lojinha e Painel Administrativo desenvolvido por Matheus M. Santos</p>  
 </footer>  
 );  
}  
  
export default Footer;  
  
  
--- FILENAME: Header.js  
import React from 'react';  
  
function Header() {  
 return (  
 <header style={{   
 position: 'fixed',   
 top: 0,   
 width: '100%',   
 height: '75px',   
 backgroundColor: '#FFB6C1',   
 display: 'flex',   
 justifyContent: 'center',   
 alignItems: 'center'   
 }}>  
 <img src="http://placekitten.com/200/300" alt="Logo" style={{ height: '80%', position: 'absolute', left: '10px' }} />  
 <h1>Mundinho Doce</h1>  
 </header>  
 );  
}  
  
export default Header;  
  
  
--- FILENAME: index.css  
body {  
 margin: 0;  
 font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', 'Roboto', 'Oxygen',  
 'Ubuntu', 'Cantarell', 'Fira Sans', 'Droid Sans', 'Helvetica Neue',  
 sans-serif;  
 -webkit-font-smoothing: antialiased;  
 -moz-osx-font-smoothing: grayscale;  
}  
  
code {  
 font-family: source-code-pro, Menlo, Monaco, Consolas, 'Courier New',  
 monospace;  
}  
  
  
--- FILENAME: index.html  
<!DOCTYPE html>  
<html lang="en">  
 <head>  
 <meta charset="utf-8" />  
 <link rel="icon" href="%PUBLIC\_URL%/favicon.ico" />  
 <meta name="viewport" content="width=device-width, initial-scale=1" />  
 <meta name="theme-color" content="#000000" />  
 <meta  
 name="description"  
 content="Web site created using create-react-app"  
 />  
 <link rel="apple-touch-icon" href="%PUBLIC\_URL%/logo192.png" />  
 <!--  
 manifest.json provides metadata used when your web app is installed on a  
 user's mobile device or desktop. See https://developers.google.com/web/fundamentals/web-app-manifest/  
 -->  
 <link rel="manifest" href="%PUBLIC\_URL%/manifest.json" />  
 <!--  
 Notice the use of %PUBLIC\_URL% in the tags above.  
 It will be replaced with the URL of the `public` folder during the build.  
 Only files inside the `public` folder can be referenced from the HTML.  
  
 Unlike "/favicon.ico" or "favicon.ico", "%PUBLIC\_URL%/favicon.ico" will  
 work correctly both with client-side routing and a non-root public URL.  
 Learn how to configure a non-root public URL by running `npm run build`.  
 -->  
 <title>React App</title>  
 </head>  
 <body>  
 <noscript>You need to enable JavaScript to run this app.</noscript>  
 <div id="root"></div>  
 <!--  
 This HTML file is a template.  
 If you open it directly in the browser, you will see an empty page.  
  
 You can add webfonts, meta tags, or analytics to this file.  
 The build step will place the bundled scripts into the <body> tag.  
  
 To begin the development, run `npm start` or `yarn start`.  
 To create a production bundle, use `npm run build` or `yarn build`.  
 -->  
 </body>  
</html>  
  
  
--- FILENAME: index.js  
import React from 'react';  
import ReactDOM from 'react-dom/client';  
import './index.css';  
import App from './App';  
import reportWebVitals from './reportWebVitals';  
  
const root = ReactDOM.createRoot(document.getElementById('root'));  
root.render(  
 <React.StrictMode>  
 <App />  
 </React.StrictMode>  
);  
  
// If you want to start measuring performance in your app, pass a function  
// to log results (for example: reportWebVitals(console.log))  
// or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals  
reportWebVitals();  
  
  
--- FILENAME: reportWebVitals.js  
const reportWebVitals = onPerfEntry => {  
 if (onPerfEntry && onPerfEntry instanceof Function) {  
 import('web-vitals').then(({ getCLS, getFID, getFCP, getLCP, getTTFB }) => {  
 getCLS(onPerfEntry);  
 getFID(onPerfEntry);  
 getFCP(onPerfEntry);  
 getLCP(onPerfEntry);  
 getTTFB(onPerfEntry);  
 });  
 }  
};  
  
export default reportWebVitals;  
  
  
--- FILENAME: robots.txt  
# https://www.robotstxt.org/robotstxt.html  
User-agent: \*  
Disallow:  
  
  
--- FILENAME: setupTests.js  
// jest-dom adds custom jest matchers for asserting on DOM nodes.  
// allows you to do things like:  
// expect(element).toHaveTextContent(/react/i)  
// learn more: https://github.com/testing-library/jest-dom  
import '@testing-library/jest-dom';