NAME: JANHVI SINGH

REGISTRATION NUMBER: 22BCE8544

Assignment 16

App.jsx

import React from 'react';

function App() {

  return (

    <div className="min-h-screen flex flex-col">

      {/\* Header \*/}

      <header className="bg-gray-800 text-white p-4 flex flex-col sm:flex-row justify-between items-center">

        <div className="text-2xl font-bold mb-2 sm:mb-0">LaunchDarkly</div>

        <nav>

          <ul className="flex flex-col sm:flex-row space-y-2 sm:space-y-0 sm:space-x-4">

            <li><a href="#" className="hover:text-blue-400">Home</a></li>

            <li><a href="#" className="hover:text-blue-400">Features</a></li>

            <li><a href="#" className="hover:text-blue-400">Pricing</a></li>

            <li><a href="#" className="hover:text-blue-400">Contact</a></li>

          </ul>

        </nav>

      </header>

      {/\* Main Content \*/}

      <main className="flex-grow p-6 bg-gray-100">

        <h1 className="text-4xl font-bold text-gray-800 mb-4">Welcome to Our Responsive Page</h1>

        <p className="text-lg text-gray-700 leading-relaxed">

          This is an example of a responsive web page built using React and Tailwind CSS.

          Tailwind's utility-first approach allows for rapid UI development with highly customizable designs.

          Observe how the layout adapts to different screen sizes.

        </p>

        <div className="mt-8">

          <button className="bg-blue-500 text-white py-2 px-4 rounded-md hover:bg-blue-600 transition duration-300 ease-in-out">

            Learn More

          </button>

        </div>

      </main>

      {/\* Footer \*/}

      <footer className="bg-gray-800 text-white p-4 text-center text-sm">

        &copy; {new Date().getFullYear()} LaunchDarkly. All rights reserved.

      </footer>

    </div>

  );

}

export default App;

README

**# React Tailwind Demo**

This project demonstrates the setup and usage of Tailwind CSS in a React application built with Vite.

**## Installation and Setup**

1.  **\*\*Create a new Vite React project:\*\***

    ```bash

    npm create vite@latest react-tailwind-demo --template react

    cd react-tailwind-demo

    ```

2.  **\*\*Install project dependencies:\*\***

    ```bash

    npm install

    ```

3.  **\*\*Install Tailwind CSS and its peer dependencies:\*\***

    ```bash

    npm install -D tailwindcss postcss autoprefixer

    ```

4.  **\*\*Initialize Tailwind CSS configuration:\*\***

    ```bash

    npx tailwindcss init

    ```

5.  **\*\*Configure Tailwind CSS to scan your template files:\*\***

    Update `tailwind.config.js` to include the following `content` array:

    ```javascript

    /\*\* @type {import('tailwindcss'].Config} \*/

    module.exports = {

      content: [

        "./index.html",

        "./src/\*\*/\*.{js,jsx,ts,tsx}",

      ],

      theme: {

        extend: {},

      },

      plugins: [],

    }

    ```

6.  **\*\*Add Tailwind's directives to your CSS:\*\***

    Add the following to your `src/index.css` file:

    ```css

    @tailwind base;

    @tailwind components;

    @tailwind utilities;

    ```

**## Styling with Tailwind CSS**

This project utilizes Tailwind CSS's utility-first classes to style various components, including the header, main content, footer, and a button.

**### Layout and Responsiveness:**

\* **\*\*Flexbox for Layout:\*\*** The `header` uses `flex`, `justify-between`, and `items-center` for horizontal alignment and spacing. The main container uses `flex flex-col` and `min-h-screen` to ensure the footer stays at the bottom.

\* **\*\*Responsive Header Navigation:\*\*** The navigation links in the header (`ul`) use `flex flex-col sm:flex-row` and `space-y-2 sm:space-y-0 sm:space-x-4`. This means:

    \* On small screens (`sm:` breakpoint and below), the links will stack vertically with `space-y-2`.

    \* On medium screens and above (`sm:` breakpoint), they will display horizontally with `space-x-4`.

\* **\*\*Padding and Margins:\*\*** Utilities like `p-4`, `p-6`, `mt-8`, `mb-4` are used for consistent spacing.

\* **\*\*Background and Text Colors:\*\*** `bg-gray-800`, `text-white`, `bg-gray-100`, `text-gray-800`, `text-gray-700` are used for styling.

**### Typography:**

\* **\*\*Titles and Paragraphs:\*\*** `text-4xl`, `font-bold`, `text-lg`, `leading-relaxed` are applied to control font size, weight, and line height.

**### Buttons:**

\* A simple button is styled with `bg-blue-500` (background color), `text-white` (text color), `py-2 px-4` (padding), and `rounded-md` (border radius).

\* **\*\*Hover Effect:\*\*** `hover:bg-blue-600` is used to change the background color when the button is hovered over, providing visual feedback. `transition duration-300 ease-in-out` is added for a smooth transition effect.

**## Running the Project**

To run the project after setup, use:

```bash

npm run dev

Screenshot

