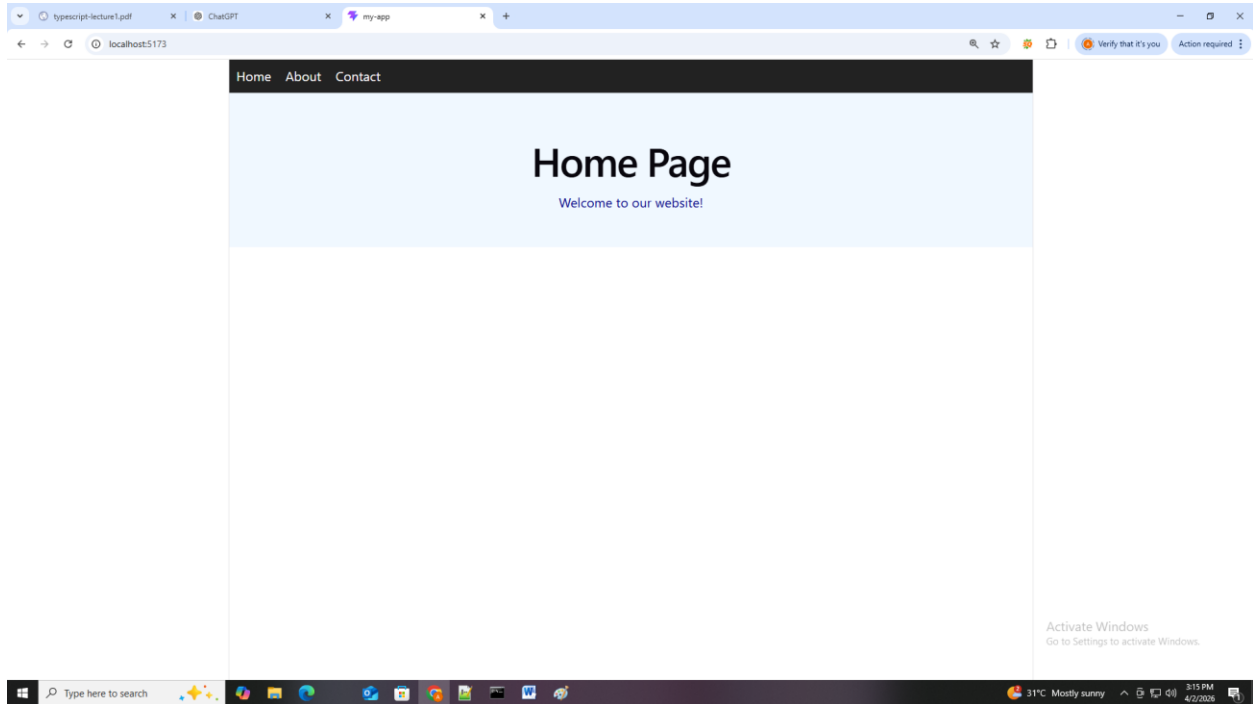


React js with vite and Typescript Project for displaying html ,about ,contact pages:-



Let's set up a **React + Vite + TypeScript** project with **Home, About, Contact** pages, **CSS styling**, and a **navbar**, all in a **clean structure**.

Here's the **complete step-by-step setup**:

## 1. Create the project

```
npm create vite@latest my-app  
cd my-app
```

- Framework: **React**
- Variant: **TypeScript**

```
D:\typescript>npm create vite@latest my-app
> npx
> create-vite my-app

|
o Select a framework:
|  React
* Select a variant:
|  > TypeScript
|     TypeScript + React Compiler
|  JavaScript
|  JavaScript + React Compiler
|  RSC
|  React Router v7 https://reactrouter.com
|  TanStack Router https://tanstack.com/router
|  RedwoodSDK https://rwsdk.com
|  Vike https://vike.dev
```

- Press y to Install dependencies:

```
npm install
```

- Run dev server:

```
npm run dev
```

---

## 2. Folder structure

```
my-app/
├── src/
│   ├── components/
│   │   ├── Navbar.tsx
│   │   └── Navbar.css
│   ├── pages/
│   │   ├── Home.tsx
│   │   ├── About.tsx
│   │   └── Contact.tsx
│   ├── App.tsx
│   ├── main.tsx
│   └── index.css
```

---

## 3. Create Pages

### src/pages/Home.tsx

```
import React from "react";
import "./Home.css";

const Home: React.FC = () => (
  <div className="page home">
    <h1>Home Page</h1>
    <p>Welcome to our website!</p>
  </div>
);

export default Home;
```

### src/pages/About.tsx

```
import React from "react";
import "./About.css";

const About: React.FC = () => (
  <div className="page about">
    <h1>About Page</h1>
    <p>We are a company that values excellence.</p>
  </div>
);

export default About;
```

### src/pages/Contact.tsx

```
import React from "react";
import "./Contact.css";

const Contact: React.FC = () => (
  <div className="page contact">
    <h1>Contact Page</h1>
    <p>Email: contact@company.com</p>
    <p>Phone: +123 456 789</p>
  </div>
);

export default Contact;
```

---

## 4. Add CSS for pages

Example: `Home.css`

```
.page.home {
  text-align: center;
  color: darkblue;
  background-color: #f0f8ff;
  padding: 50px;
}
```

`About.css`

```
.page.about {
  text-align: center;
  color: darkgreen;
  background-color: #f5fff0;
  padding: 50px;
}
```

`Contact.css`

```
.page.contact {
  text-align: center;
  color: darkred;
  background-color: #fff0f0;
  padding: 50px;
}
```

---

## 5. Create Navbar

`src/components/Navbar.tsx`

```
import React from "react";
import { Link } from "react-router-dom";
import "./Navbar.css";

const Navbar: React.FC = () => (
  <nav className="navbar">
    <Link to="/">Home</Link>
    <Link to="/about">About</Link>
    <Link to="/contact">Contact</Link>
  </nav>
);

export default Navbar;
```

### Navbar.css

```
.navbar {
  display: flex;
  gap: 20px;
  background-color: #222;
  color: white;
  padding: 10px;
}

.navbar a {
  color: white;
  text-decoration: none;
}

.navbar a:hover {
  text-decoration: underline;
}
```

---

## 6. Set up App.tsx

### src/App.tsx

```
import React from "react";
import { BrowserRouter, Routes, Route } from "react-router-dom";
import Navbar from "../components/Navbar";
import Home from "../pages/Home";
import About from "../pages/About";
import Contact from "../pages/Contact";

const App: React.FC = () => {
  return (
    <BrowserRouter>
      <Navbar />
      <Routes>
        <Route path="/" element={<Home />} />
        <Route path="/about" element={<About />} />
        <Route path="/contact" element={<Contact />} />
      </Routes>
    </BrowserRouter>
  );
};

export default App;
```

---

## 7. Entry point main.tsx

src/main.tsx

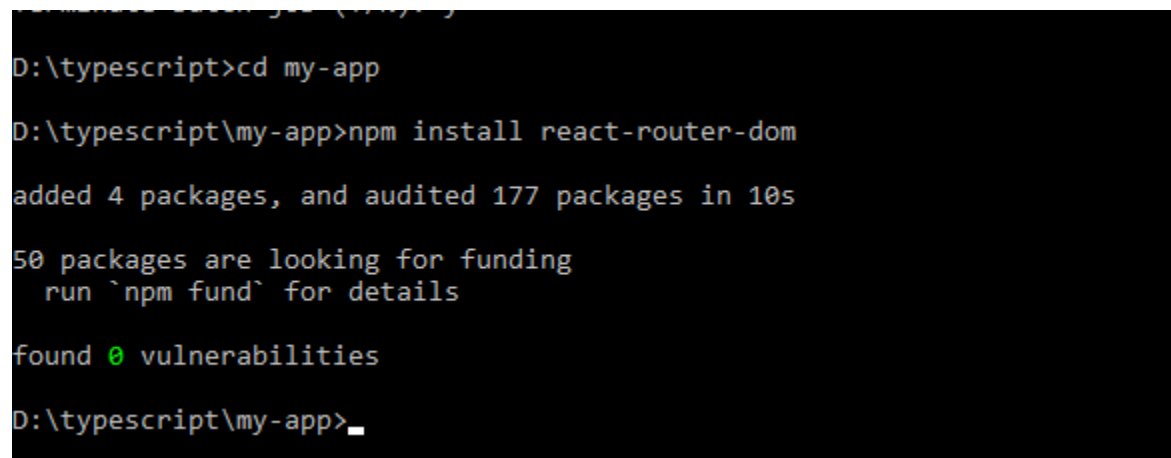
```
import React from "react";
import ReactDOM from "react-dom/client";
import App from "./App";
import "./index.css";

ReactDOM.createRoot(document.getElementById("root")!).render(
  <React.StrictMode>
    <App />
  </React.StrictMode>
);
```

---

## 8. Install React Router

npm install react-router-dom



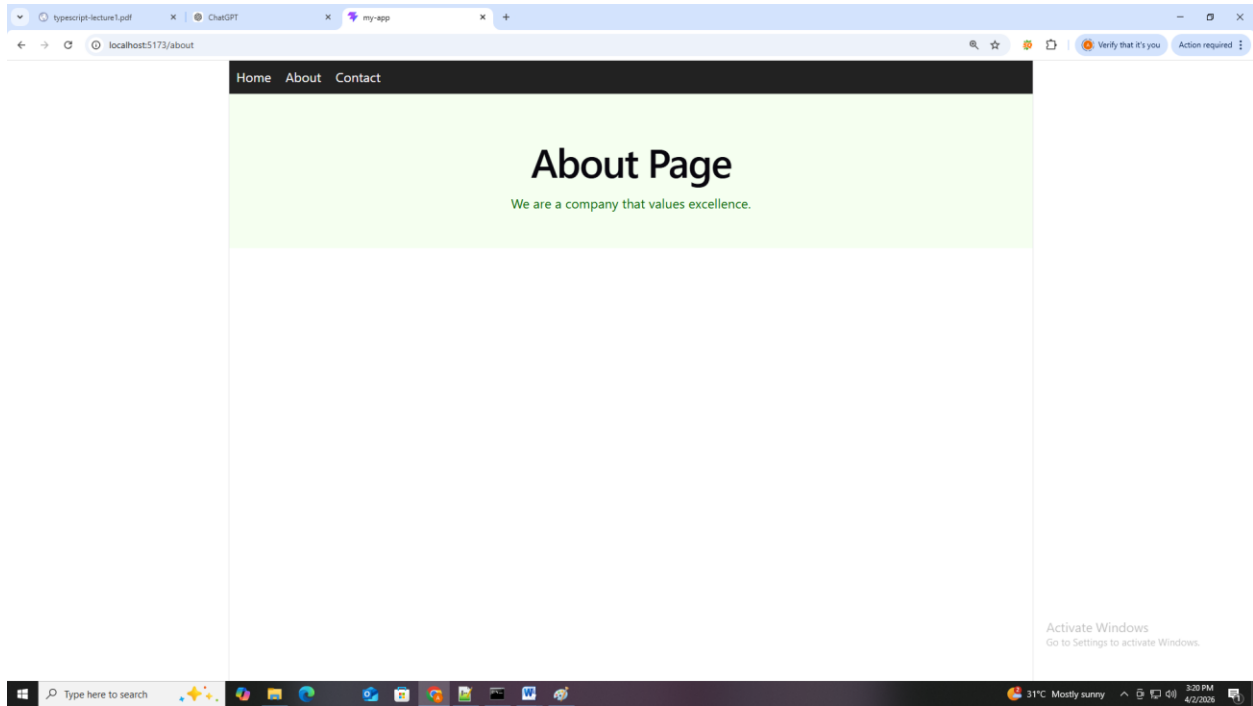
```
D:\typescript>cd my-app
D:\typescript\my-app>npm install react-router-dom
added 4 packages, and audited 177 packages in 10s
50 packages are looking for funding
  run `npm fund` for details
found 0 vulnerabilities
D:\typescript\my-app>_
```

## 9. Run the app

npm run dev

- Open the browser → <http://localhost:5173>
- You now have:
  - Home, About, Contact pages
  - Navigation bar
  - Styled pages
  - Full **React + TypeScript + Vite** setup

Output:-



Let's enhance the **About** page with a **counter** using **React useState**. You'll have **Increase** and **Decrease** buttons.

---

## 1. Update **About.tsx**

```
import React, { useState } from "react";
import "../About.css";

const About: React.FC = () => {
  // Step 1: Declare a state variable
  const [count, setCount] = useState<number>(0);

  // Step 2: Handlers
  const increase = () => setCount(count + 1);
  const decrease = () => setCount(count - 1);

  return (
    <div className="page about">
      <h1>About Page</h1>
      <p>We are a company that values excellence.</p>

      <div className="counter">
        <h2>Counter: {count}</h2>
        <button onClick={increase}>Increase</button>
        <button onClick={decrease}>Decrease</button>
      </div>
    </div>
  );
};

export default About;
```

---

## 2. Add CSS for Counter (**About.css**)

```
.page.about .counter {
  margin-top: 30px;
  text-align: center;
}

.page.about .counter button {
  margin: 0 10px;
  padding: 10px 20px;
  font-size: 16px;
  cursor: pointer;
}

.page.about .counter h2 {
  margin-bottom: 15px;
}
```

---

## 3. How It Works

1. `useState<number>(0)` → Creates a **state variable** `count` starting at 0
2. `setCount` → Updates the state
3. Clicking **Increase** → `count + 1`
4. Clicking **Decrease** → `count - 1`
5. The page automatically **re-renders** with the new count

Output will be like this :-

