Concept of Routing in Next.js

What is Routing:

Routing in Next.js refers to the mechanism of defining and managing navigation within a web application. It allows you to define different URLs (routes) for different parts of your application and specify what content or component should be rendered when a particular URL is accessed.

Purpose of Routing:

The purpose of routing in Next.js is to enable users to navigate between different pages or views of your application seamlessly. It helps create a structured and organized user experience by allowing users to access specific content or functionality based on the URL they visit.

In Next.js, routing can be achieved using the built-in routing system provided by the framework. This system is based on the file-based routing approach, where each page of your application corresponds to a separate file in the pages directory. When a user visits a specific URL, Next.js matches that URL to the corresponding page file and renders its content

Routing Conventions:

All routes must be placed inside the app folder
Every file that corresponds to a route must be named page.tsx
Every folder corresponds to a path segment in the browser URL.

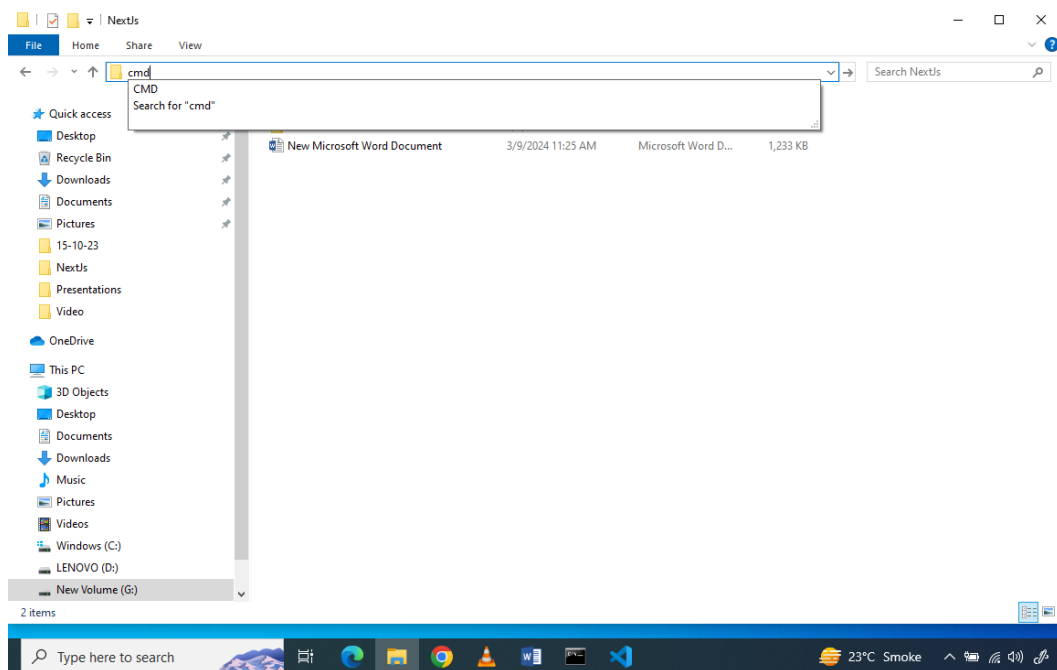# Practical Implementation of Routing in a Next.js Application

Requirements:

Node.js installed on your system.
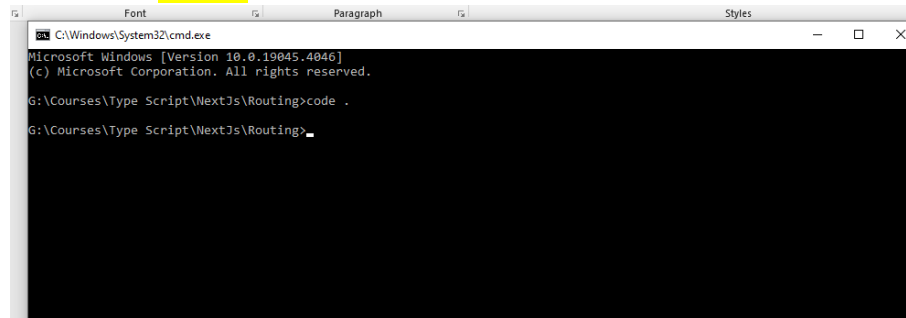A text editor such as Visual Studio Code.

Step 1: Setting Up the Project Environment

Create a new folder for your Next.js project and run the cmd command in address bar:

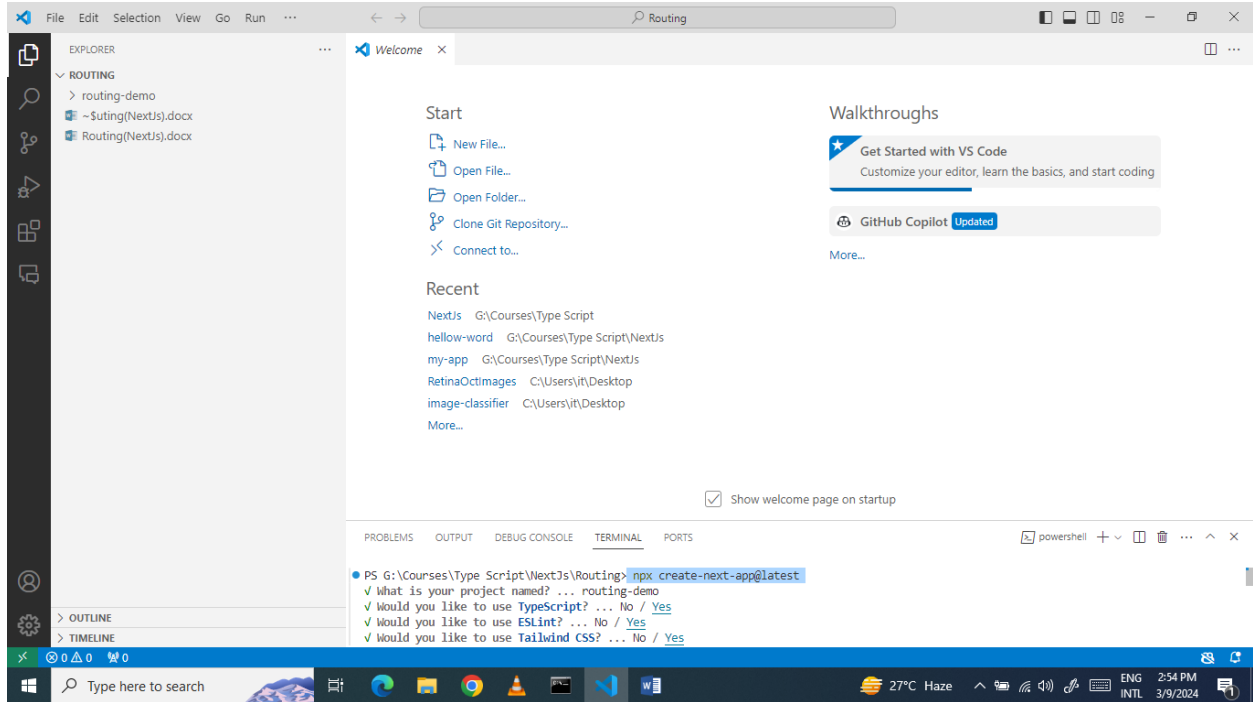Open Visual Studio Code in this folder:
by run the command code .

Step 2: Initializing Your Next.js App

With your text editor open and the project folder selected, ensure you have Node.js installed on your system.
In the terminal, run the following command to create a new Next.js app:
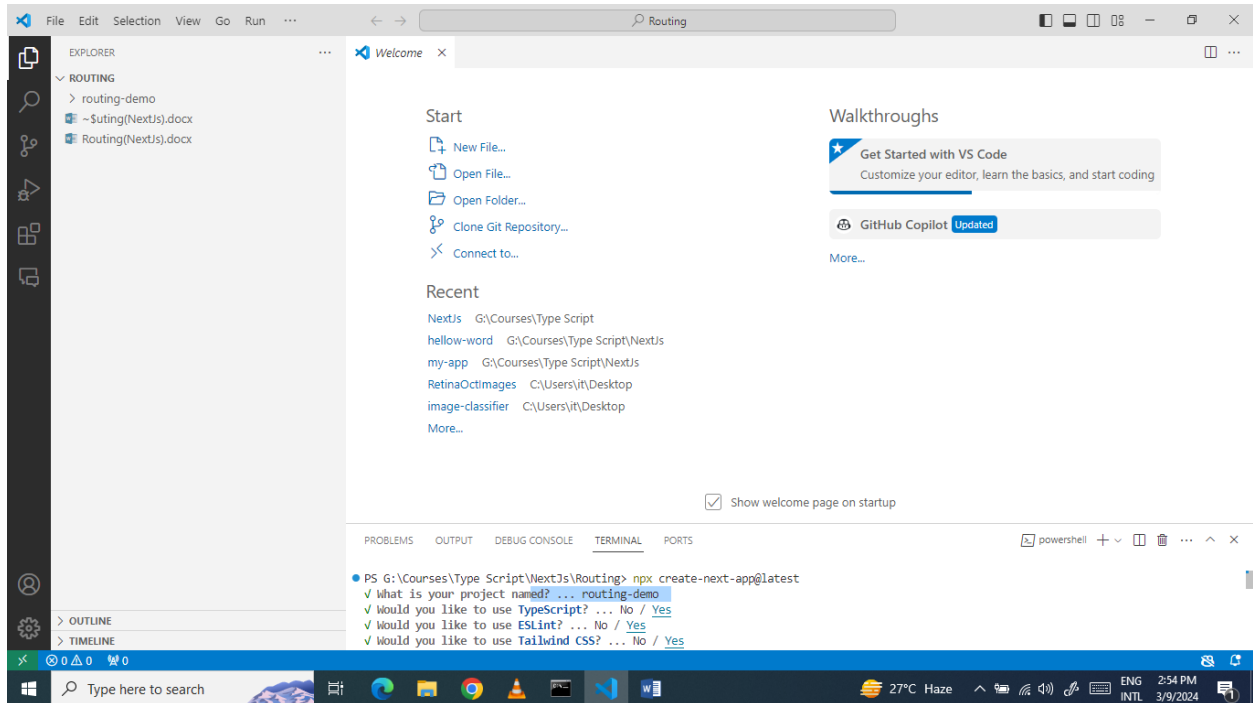npx create-next-app@latest

After running the npx create-next-app@latest command to set up a new Next.js project, you may encounter a series of prompts to configure your project.

After running the command, you will be prompted to provide the name of the app. For the subsequent prompts, you can select the default option for the remaining settings.

Name of the app: You should provide a name for your Next.js application. This name will be used as the directory name for your project. You can choose any name you like, but it's recommended to use a descriptive and relevant name for your project.
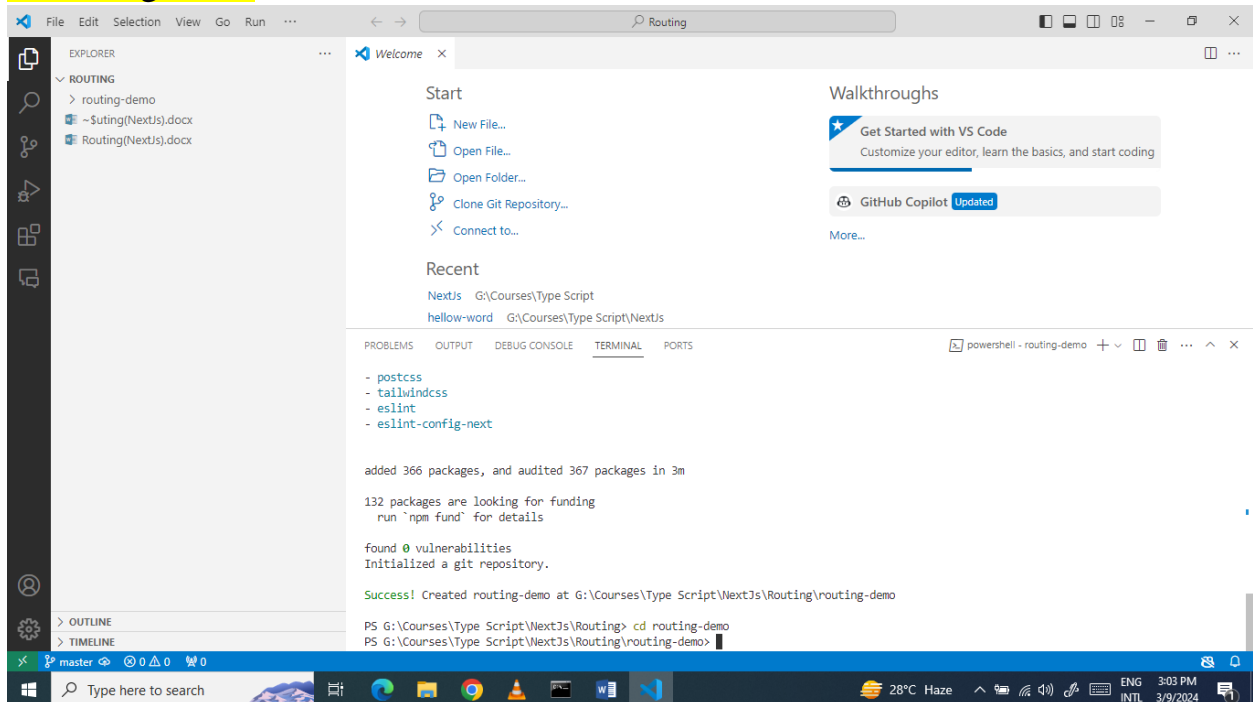
For the subsequent prompts, you can select the default option for the remaining settings.

Step 3: Running Your Next.js App

Once the installation is complete, navigate into your project directory if you're not already there:
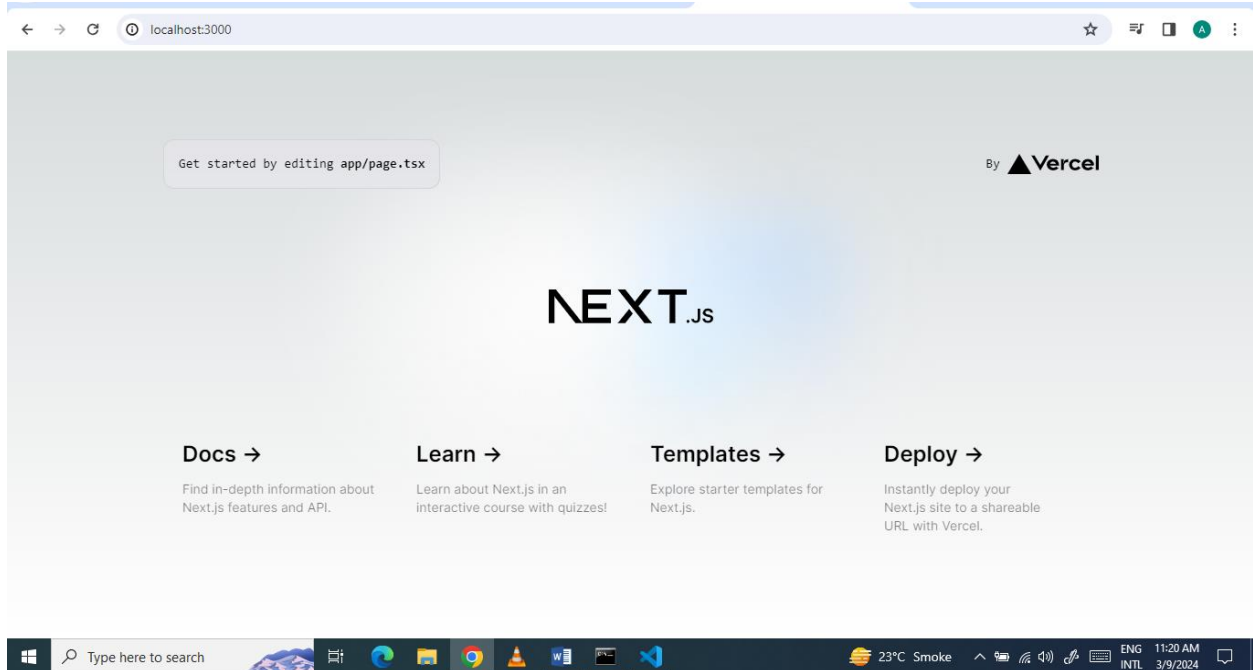
cd routing-demo

Start the development server by running the following command:
<mark>npm run dev</mark>

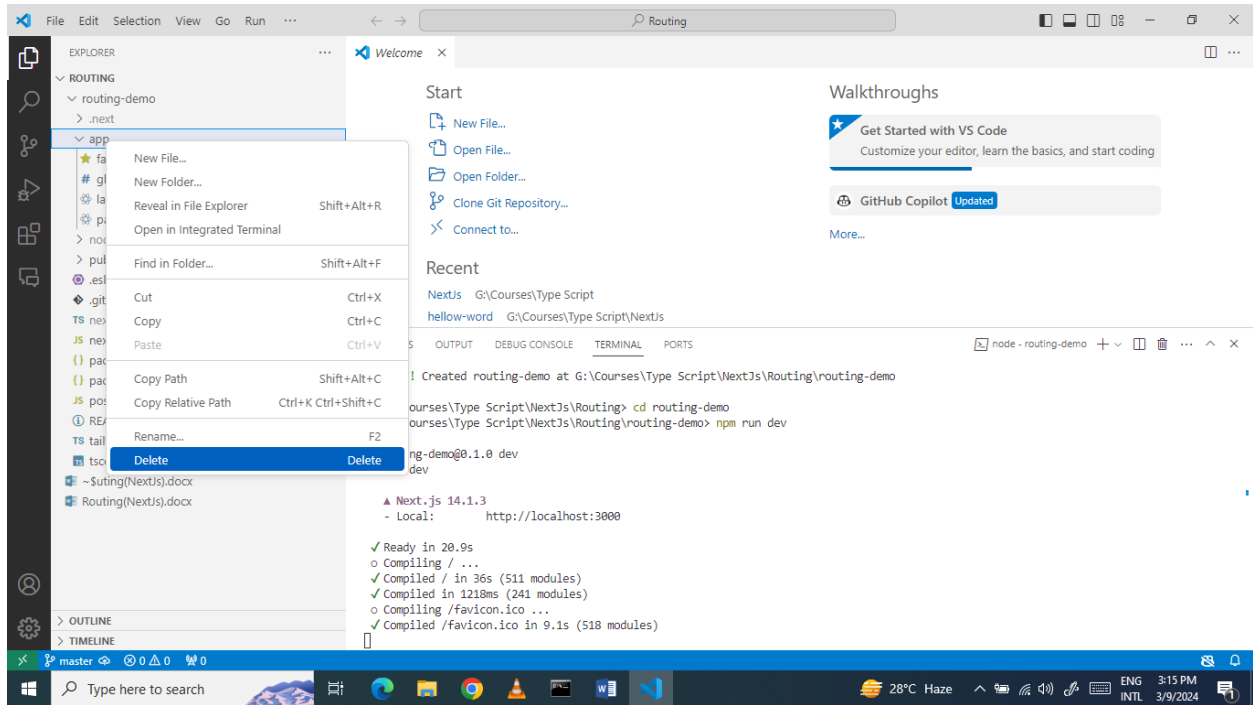This will launch your Next.js application locally. You can access it by opening your web browser and navigating to http://localhost:3000.

Step 4: Exploring Your Next.js App

With your Next.js app running, you can start exploring the project structure and making changes to the code.
Open the project folder in Visual Studio Code and begin editing files within the pages directory to create your app's pages and components.
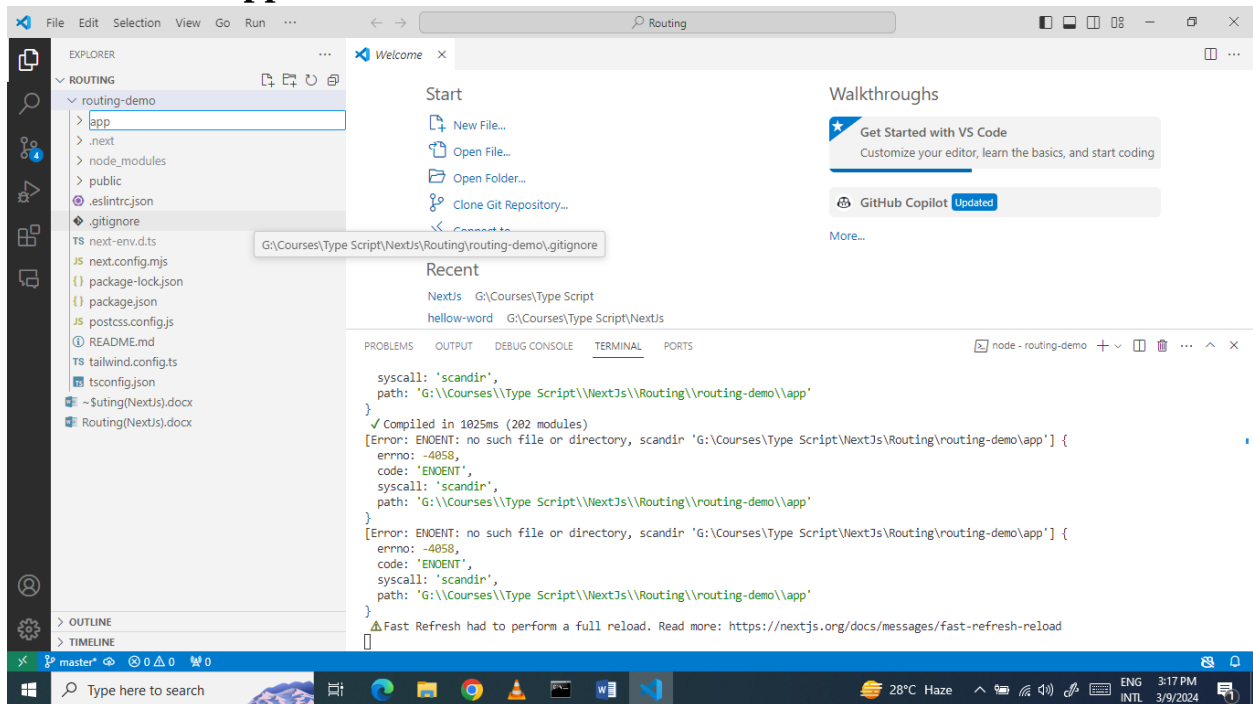
For example

If you need to recreate the app folder from scratch and then create a new page.tsx file, you can follow these steps:
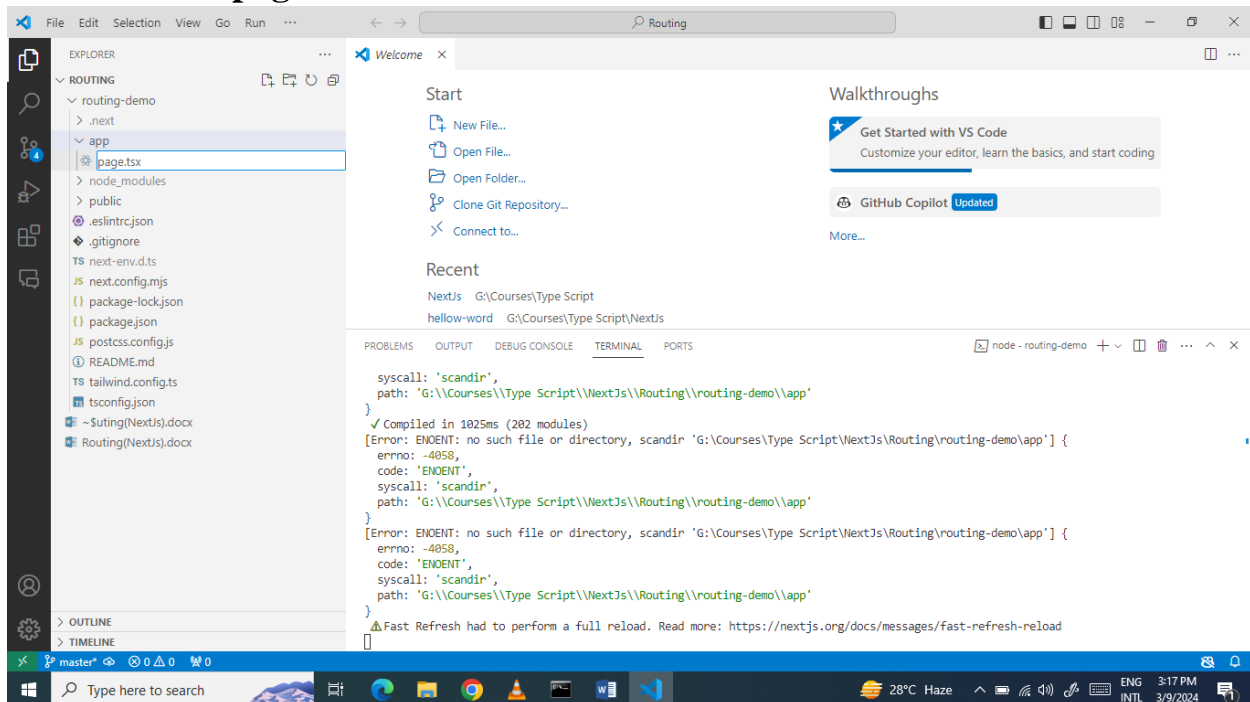Delete the Existing App Folder:

Recreate the App Folder:
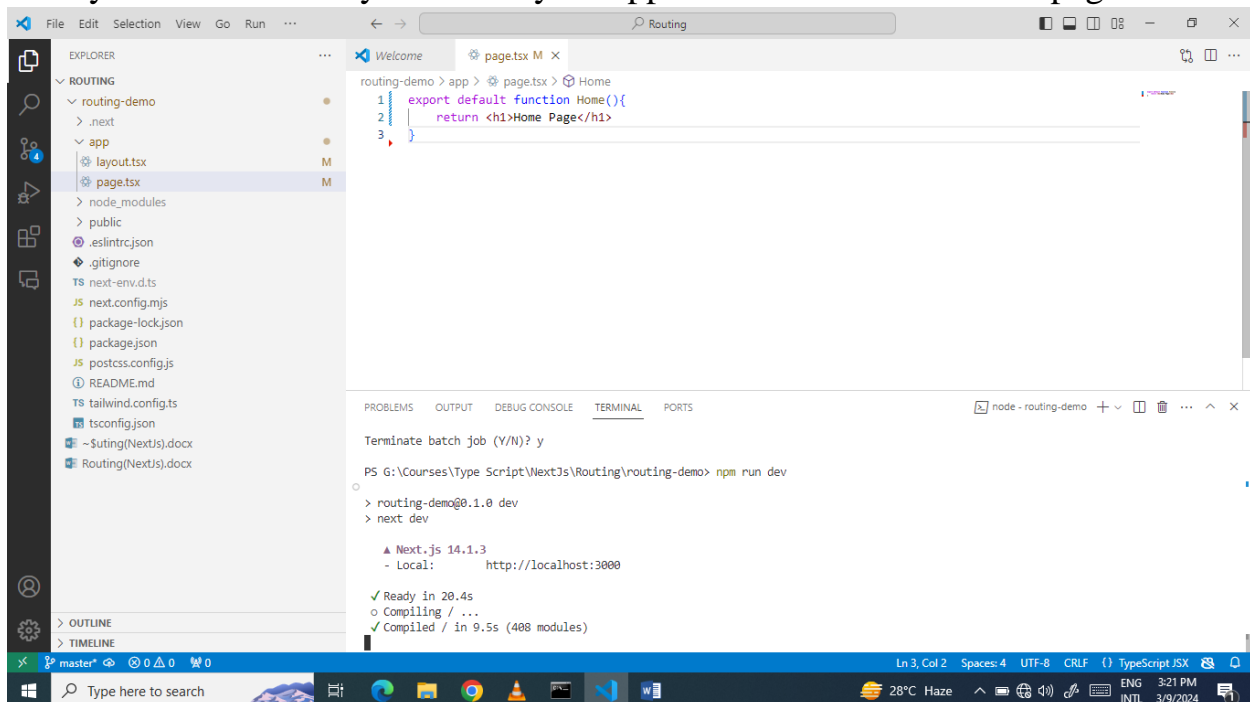
Create a New page.tsx File:
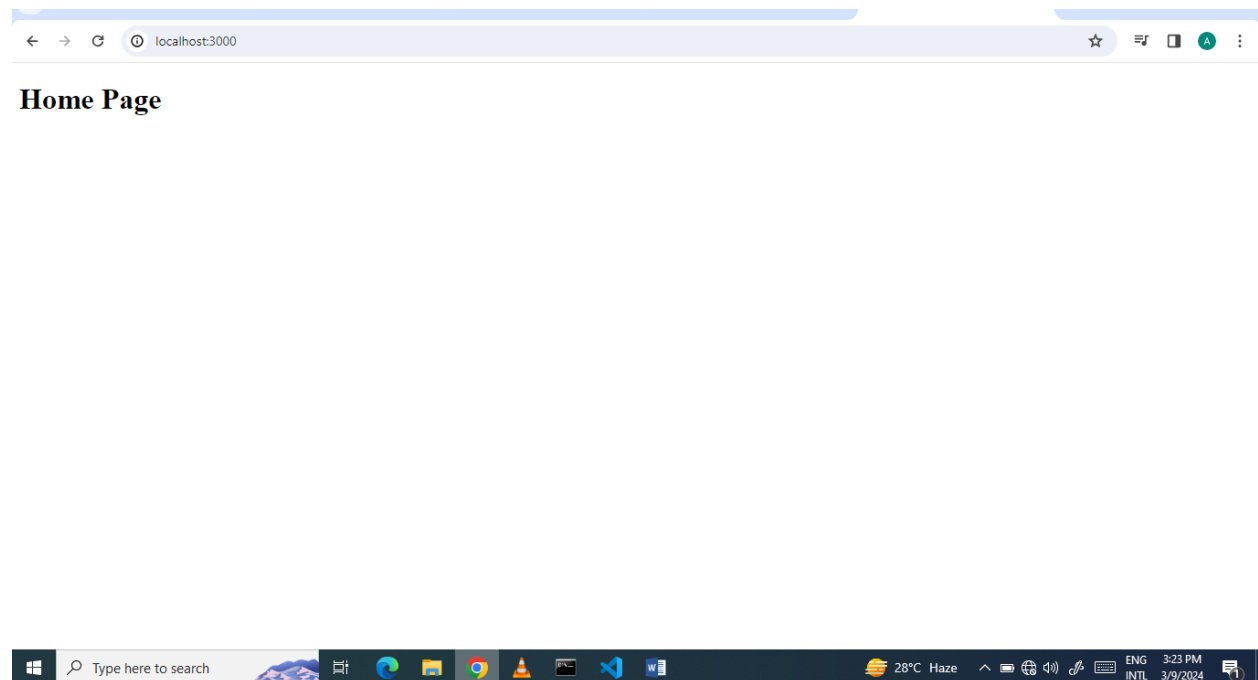

Edit the page.tsx File:

Save Changes:

Save the changes made to the page.tsx file.

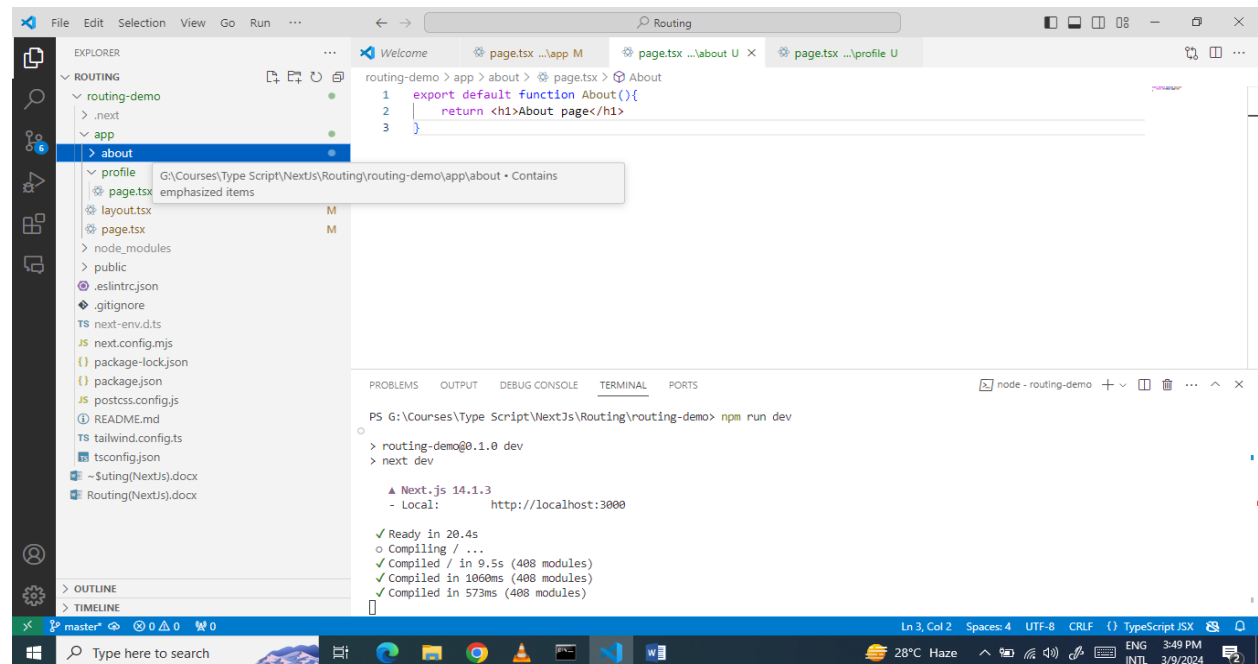Now you've successfully recreated your app folder and created a new page.tsx file.
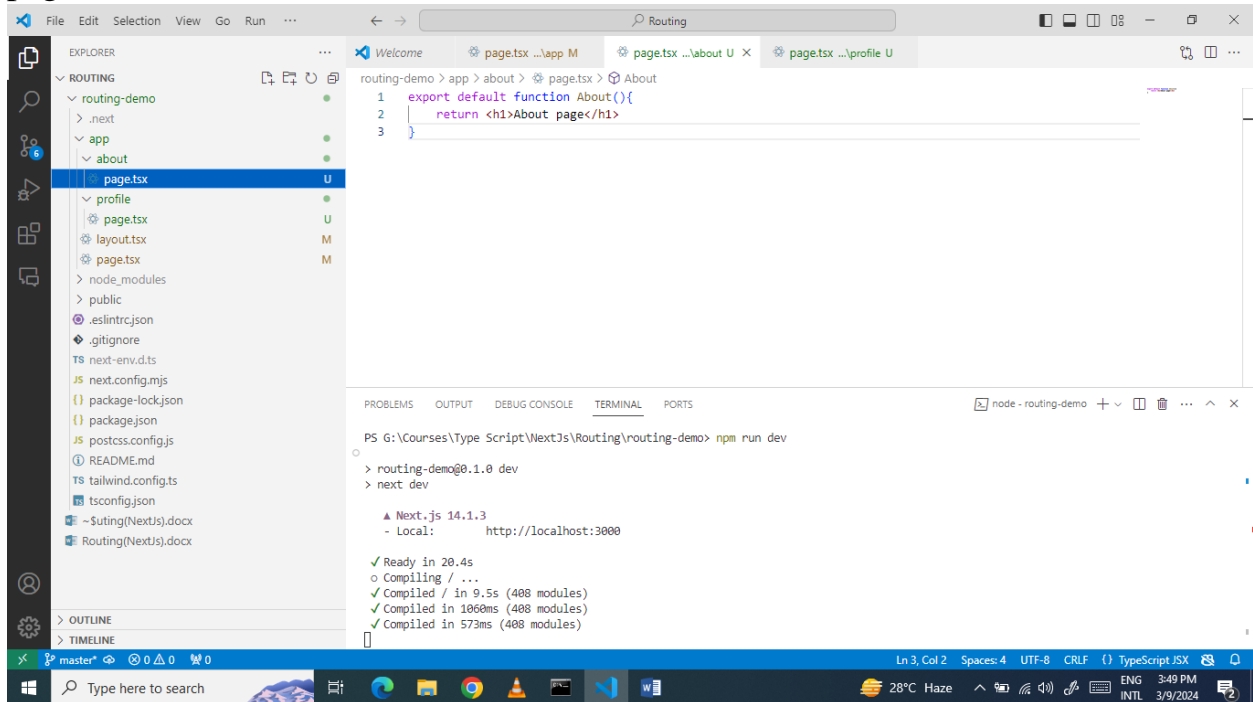
Output will be

Let's continue building your Next.js app by adding more pages and components. Here's how you can create a folder named about inside the pages directory and create a page.tsx file within the about folder:
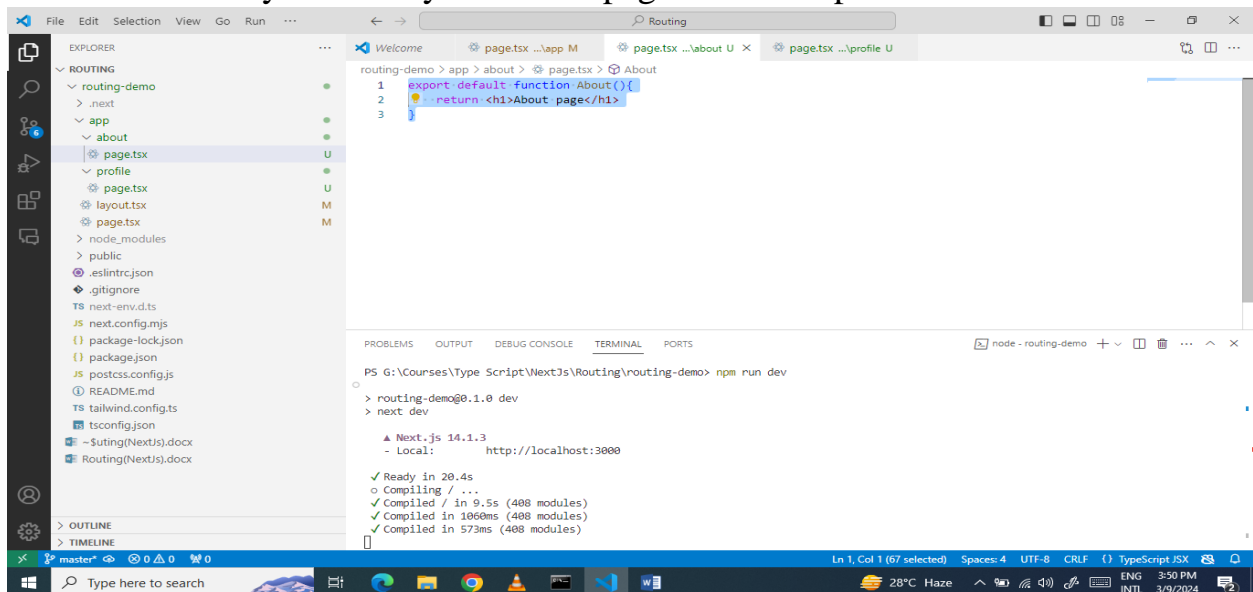
Create the About Folder

Create a New page.tsx File in the About Folder:

Now that you're inside the about folder, create a new TypeScript file named page.tsx.

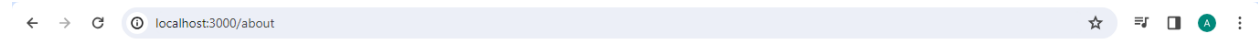## Edit the page.tsx File:

Open the page.tsx file located inside the pages/about folder in your text editor.

Add the necessary code for your About page. For example:

Save Changes:

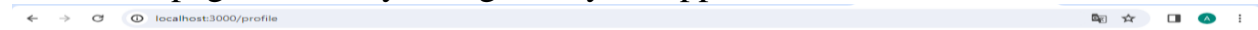Save the changes made to the page.tsx file.
Now you have successfully created a new page named About in your Next.js app.
You can access this page by navigating to /about in your browser once the
development server is running (npm run dev).

Feel free to continue adding more pages and components to your Next.js app as
needed. You can follow a similar process to create additional folders and files
within the pages directory to organize your app's structure.