

# Routes Handler in Next.js

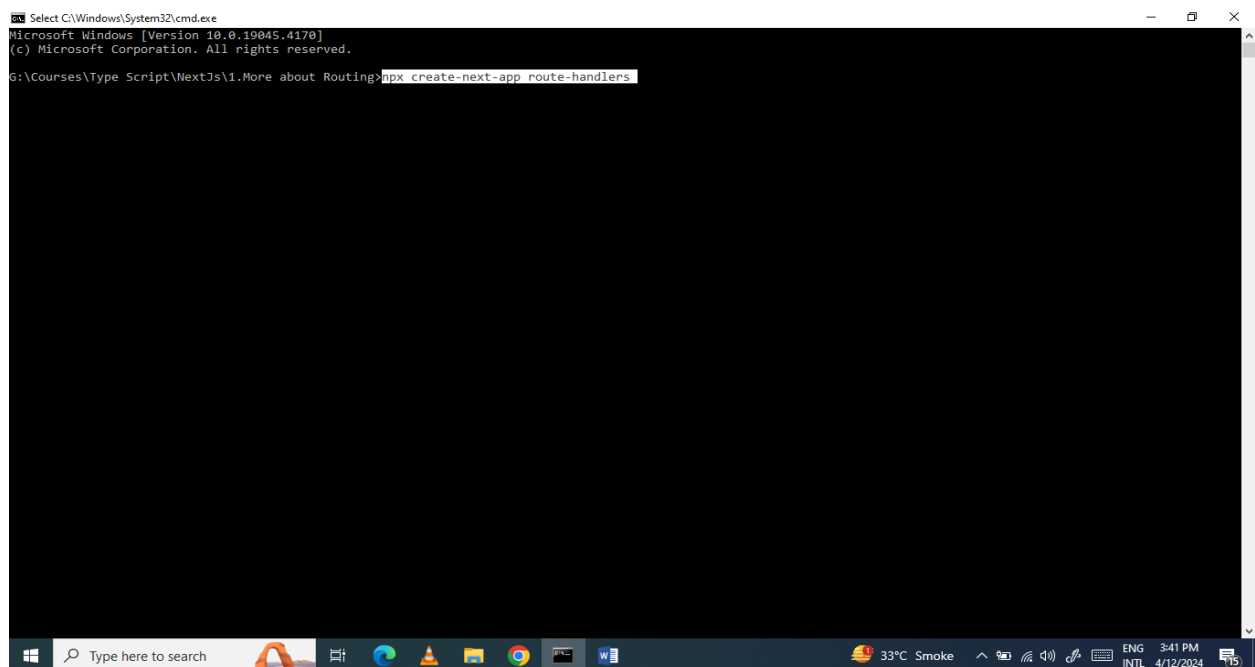
## Routes Handler in Next.js:

In Next.js, a route handler is a function or method that is responsible for handling requests to specific routes within your application. These route handlers define what content or behavior should be served when a user accesses a particular URL.

The purpose of route handlers in Next.js is to provide a way to structure and organize your application's code based on different routes or pages. Each route handler typically corresponds to a specific page or endpoint within your application.

Route handlers in Next.js can handle various tasks such as fetching data, rendering components, handling user input, and more. Unlike page routes that return HTML content, route handlers empower you to establish RESTful endpoints, granting you complete authority over the response. This enables you to execute CRUD (Create, Read, Update, and Delete). operations with a database, resembling the functionality of a Node.js and Express application, but without the need to set up and manage a separate server..

Create new app with command `npx create-next-app route-handlers`



```
Microsoft Windows [Version 10.0.19045.4170]
(c) Microsoft Corporation. All rights reserved.

G:\Courses\Type Script\NextJs\1.More about Routing>npx create-next-app route-handlers
```

```
C:\Windows\System32\cmd.exe
Need to install the following packages:
create-next-app@14.2.0
Ok to proceed? (y)
✓ Would you like to use TypeScript? ... No / Yes
✓ Would you like to use ESLint? ... No / Yes
✓ Would you like to use Tailwind CSS? ... No / Yes
✓ Would you like to use `src/` directory? ... No / Yes
✓ Would you like to use App Router? (recommended) ... No / Yes
✓ Would you like to customize the default import alias (@/*)? ... No / Yes
Creating a new Next.js app in G:\Courses\Type Script\NextJs\1.More about Routing\route-handlers.

Using npm.

Initializing project with template: app-tw

Installing dependencies:
- react
- react-dom
- next

Installing devDependencies:
- typescript
- @types/node
- @types/react
- @types/react-dom
- postcss
- tailwindcss
- eslint
- eslint-config-next

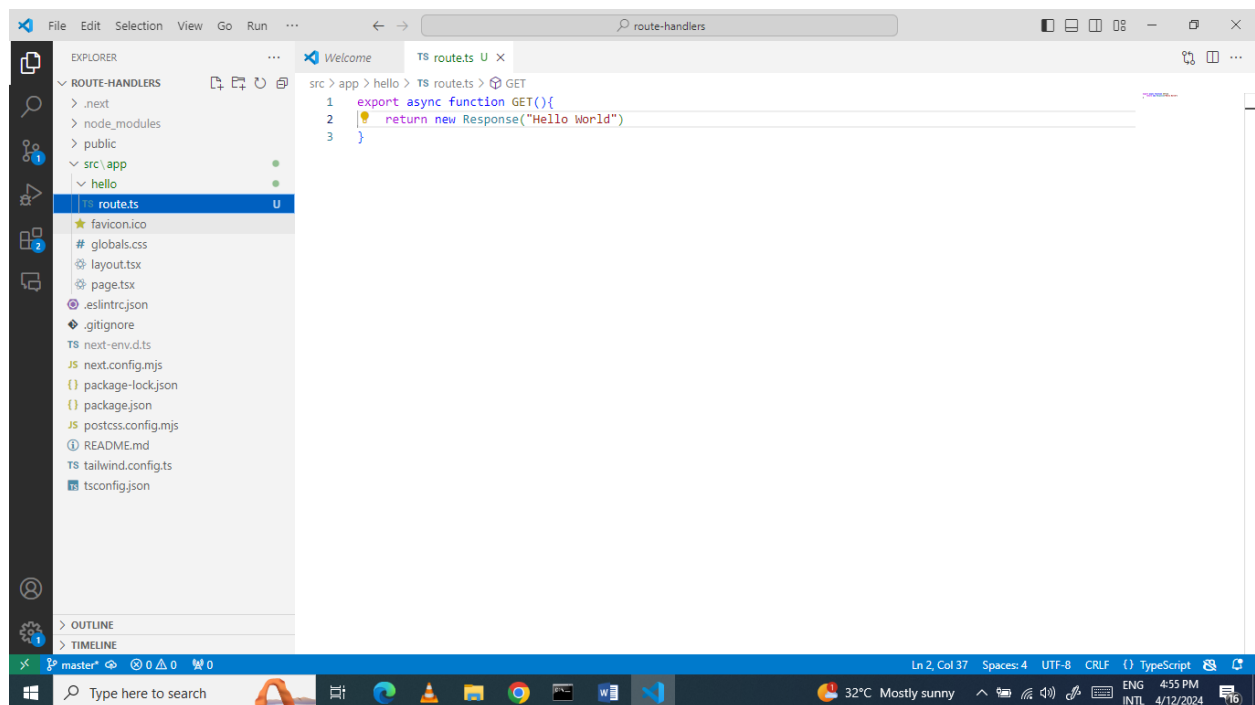
added 360 packages, and audited 361 packages in 4m
133 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
Initialized a git repository.

Success! Created route-handlers at G:\Courses\Type Script\NextJs\1.More about Routing\route-handlers

G:\Courses\Type Script\NextJs\1.More about Routing>
```

Create a folder named "hello" inside the "app" folder, and within the "hello" folder, create a file named "route.ts".



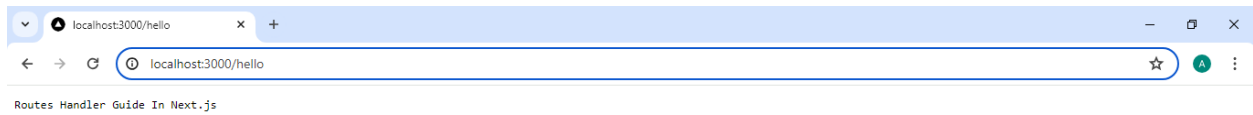
The screenshot shows the Visual Studio Code interface. On the left, the Explorer sidebar displays the project structure. The 'src/app' directory is expanded, showing a new 'hello' folder. Inside 'hello', the 'route.ts' file is selected. The main editor area shows the content of 'route.ts', which contains a single GET route handler. The status bar at the bottom indicates the file is in 'TypeScript' mode.

```
src > app > hello > TS route.ts > GET
1  export async function GET() {
2    return new Response("Hello World")
3  }
```

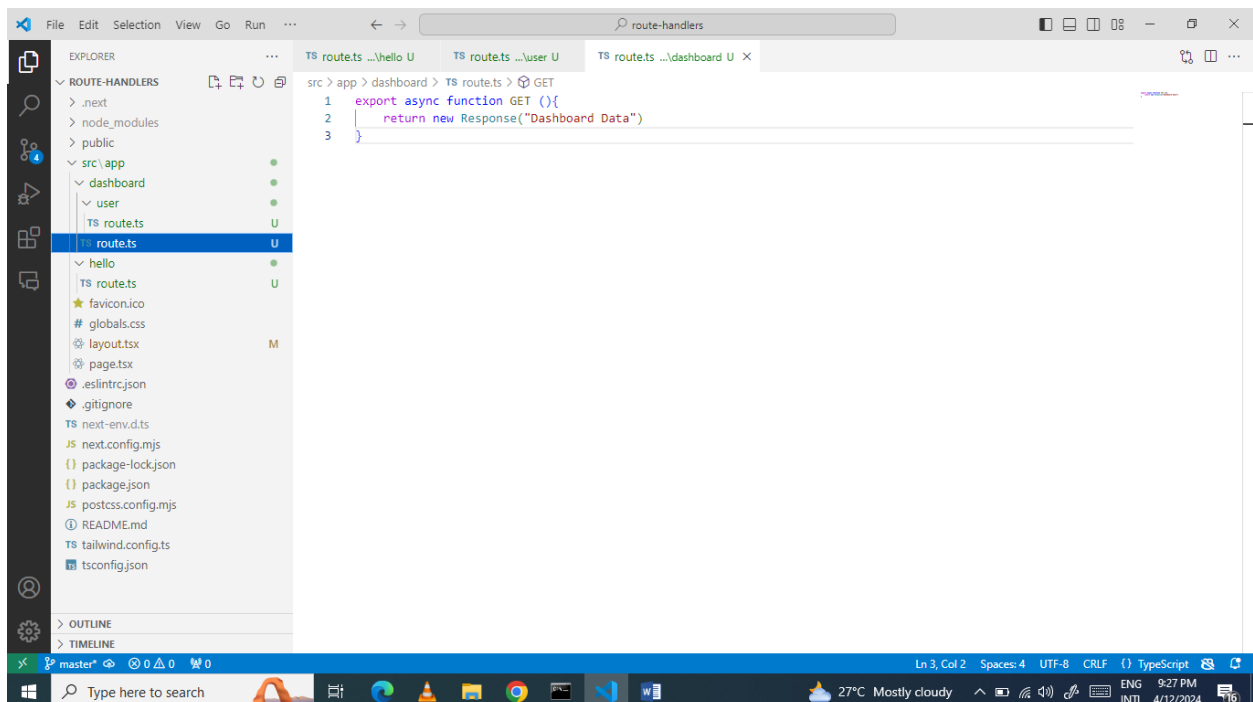
Run the command

**npm run dev**

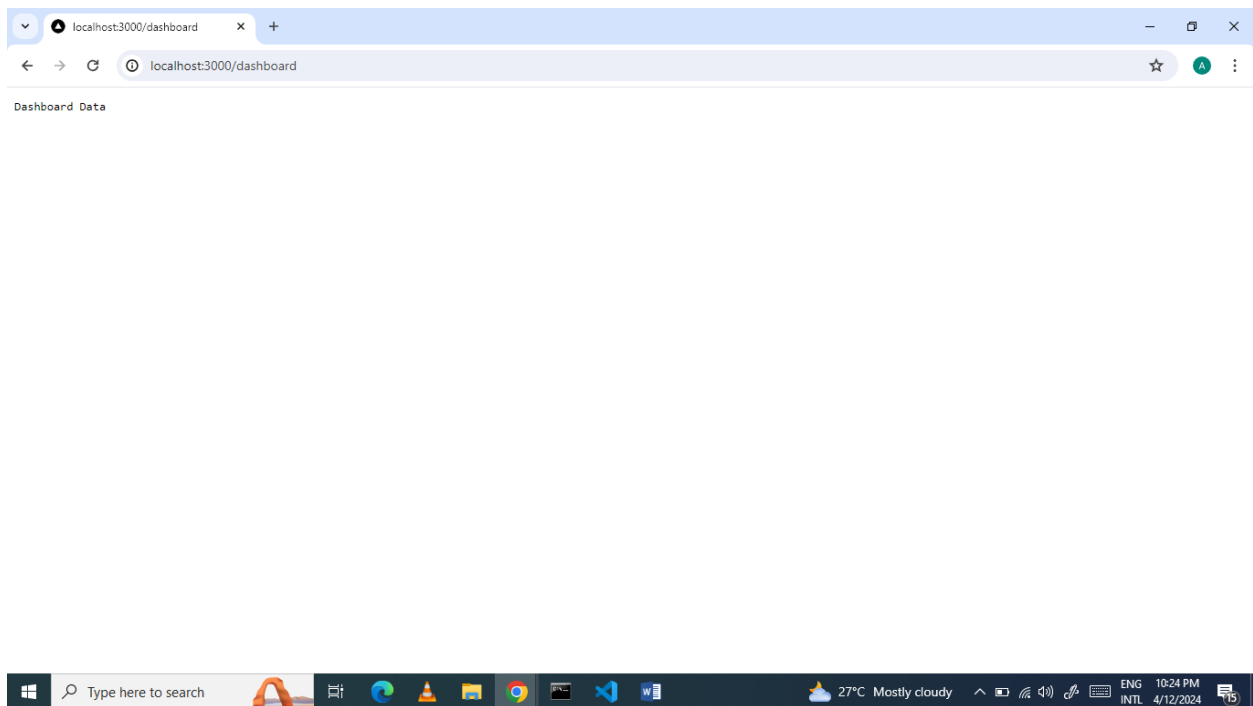
"Go to your browser and navigate to <http://localhost:3000/hello>. The output will be as shown in the screenshots below."



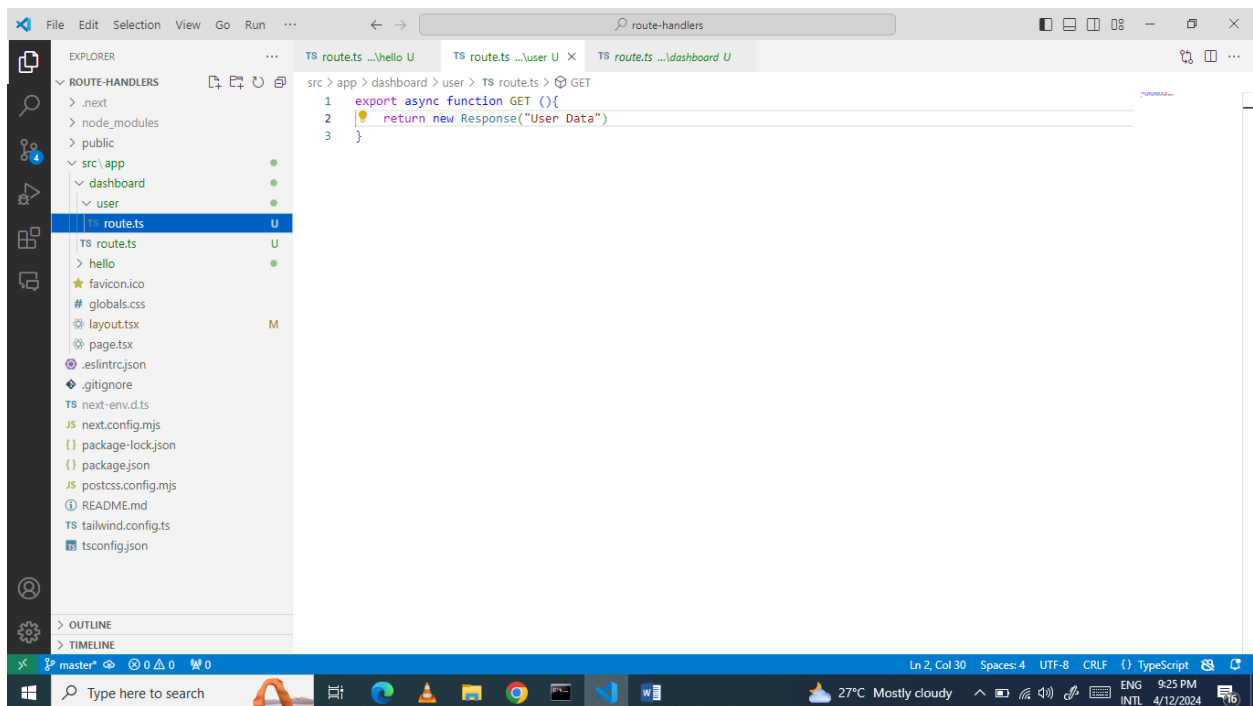
Create a dashboard folder inside the app folder. Inside the dashboard folder, create a route.ts file.



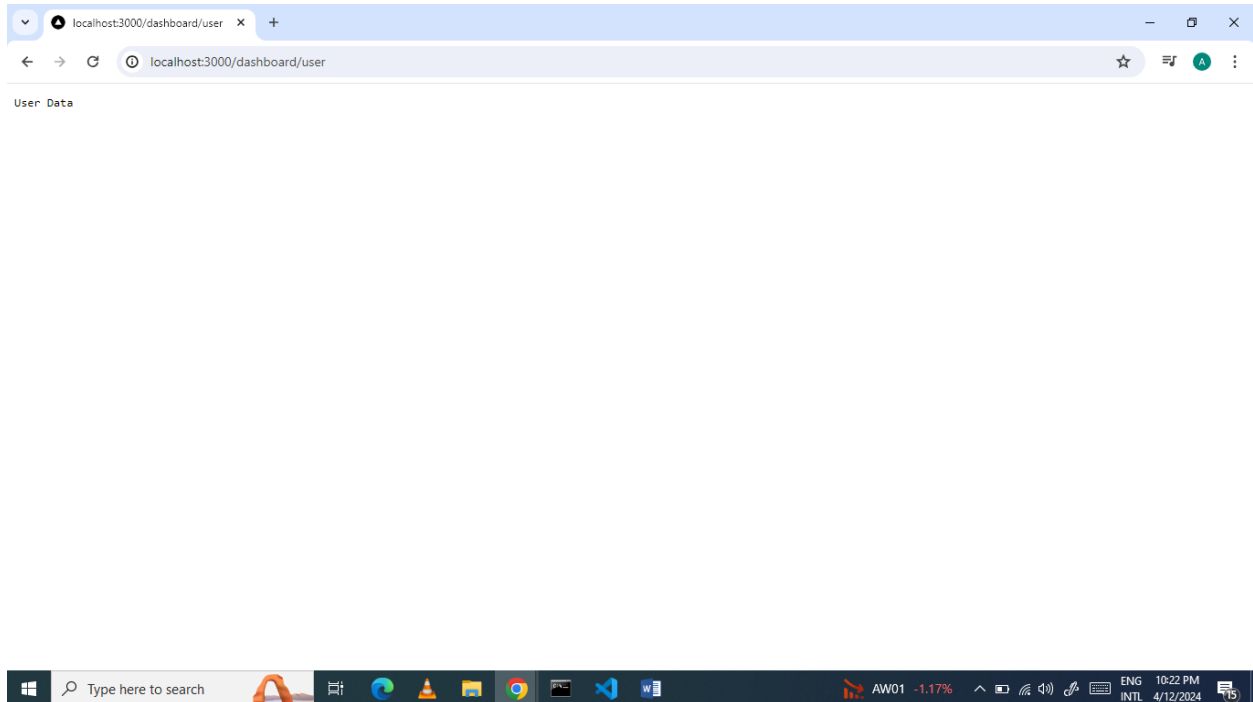
"Go to the browser and navigate to <http://localhost:3000/dashboard>. The output will be as shown in the screenshots below."



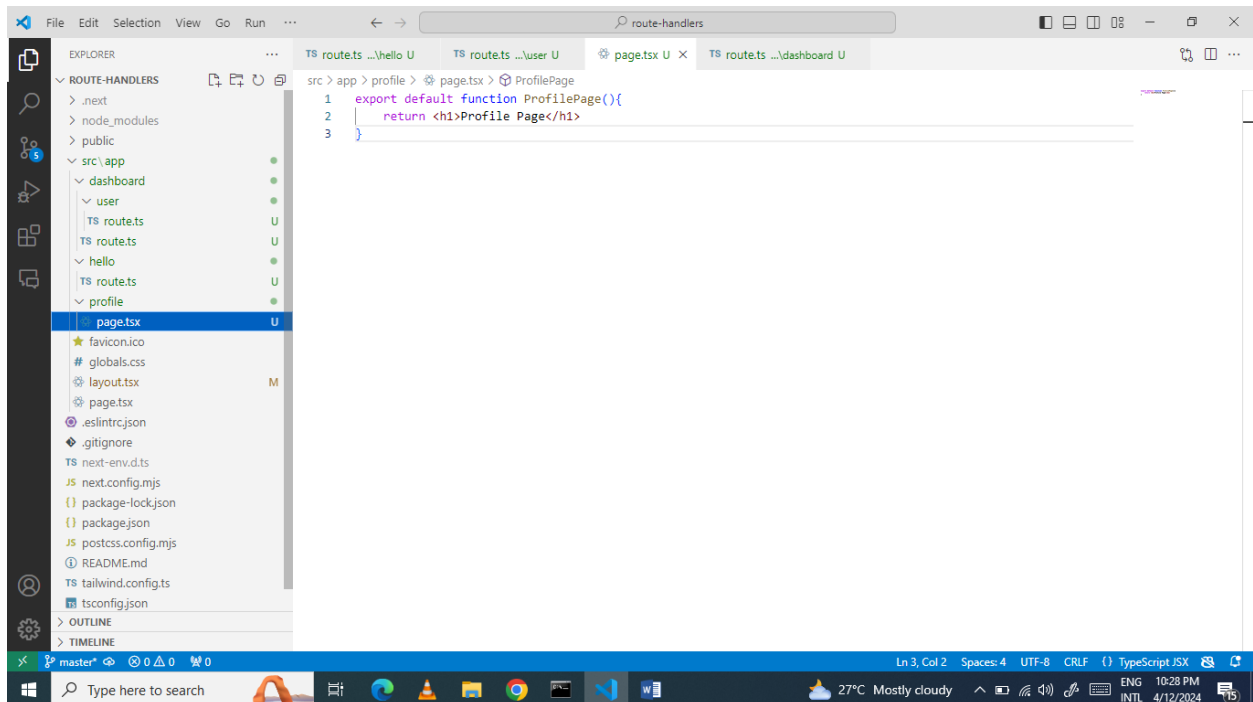
Create a user folder inside the dashboard folder. Inside the user folder, create a route.ts file.



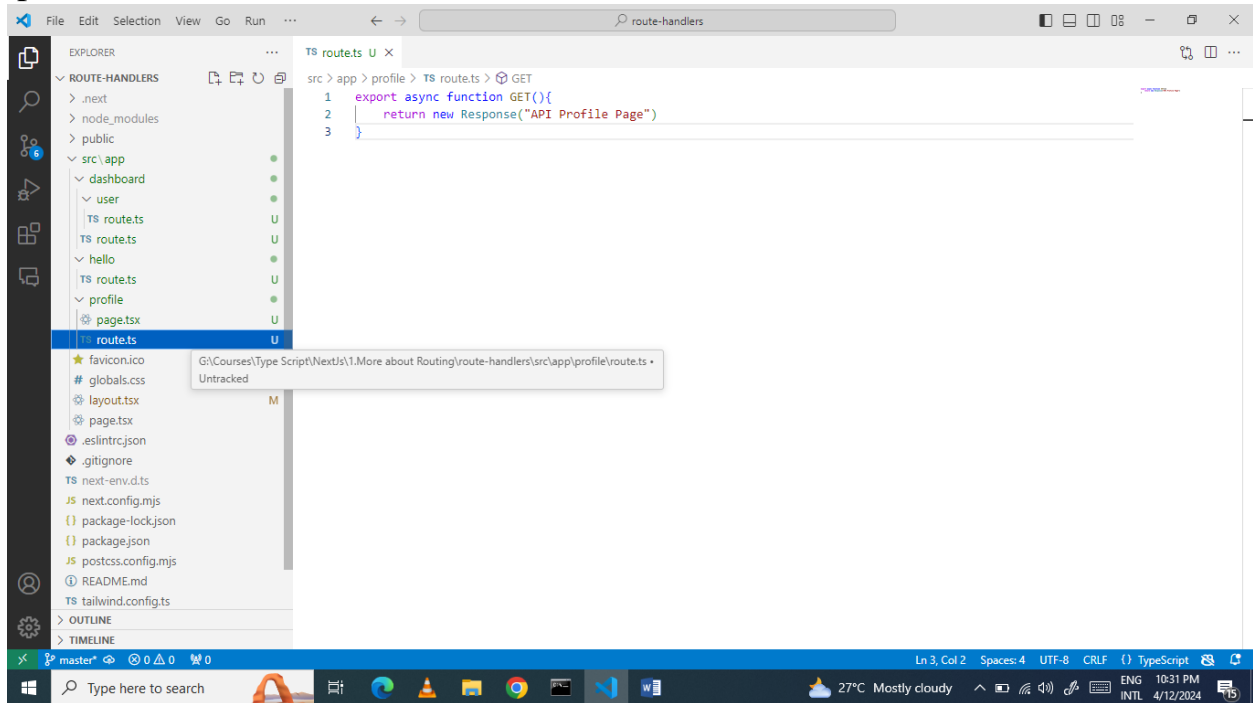
"Go to your browser and navigate to <http://localhost:3000/dashboard/user>. The output will be as shown in the screenshots below."



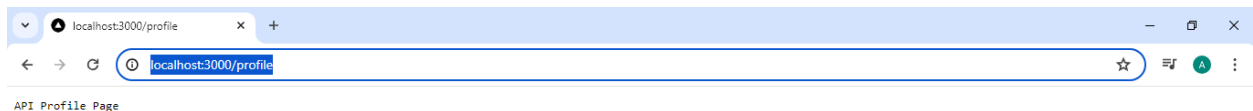
Create a profile folder inside the app folder and create page.tsx and route.ts files inside the profile folder.



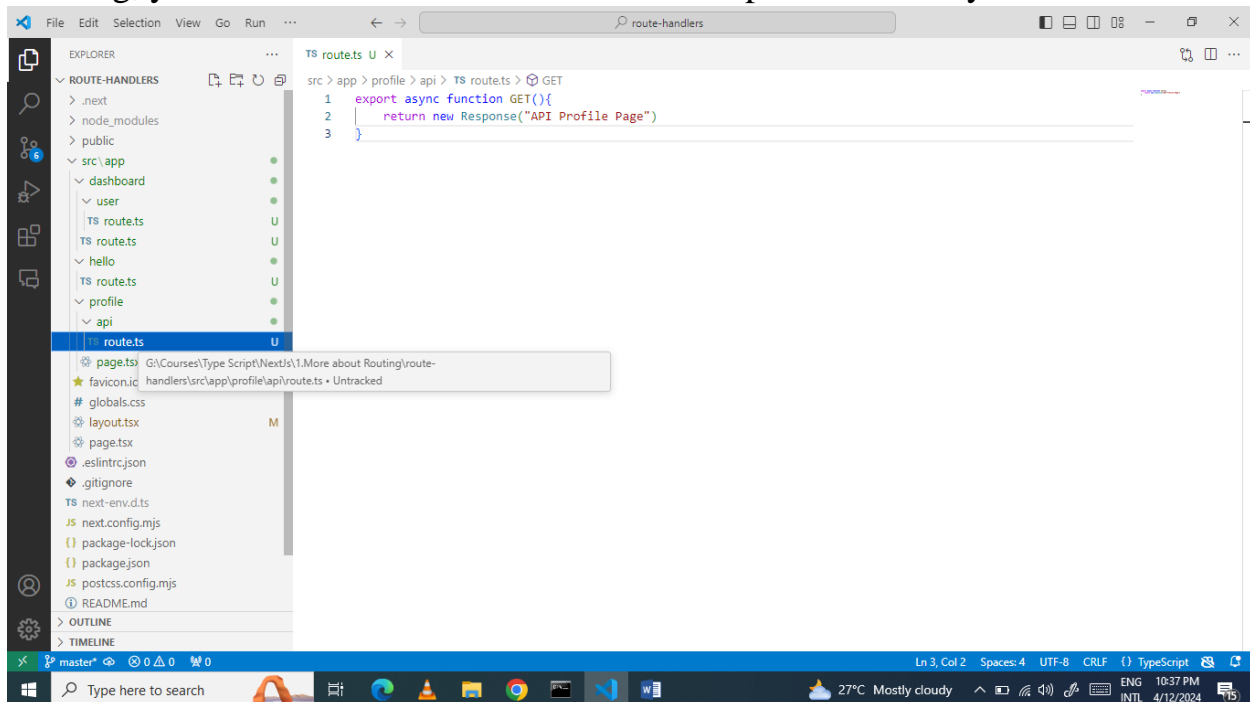
"Create a route.ts file inside the profile page. With a handler function "profile API data", this route.ts will handle the request by default. Therefore, if we navigate to /profile, we will see the text "Profile API data"."



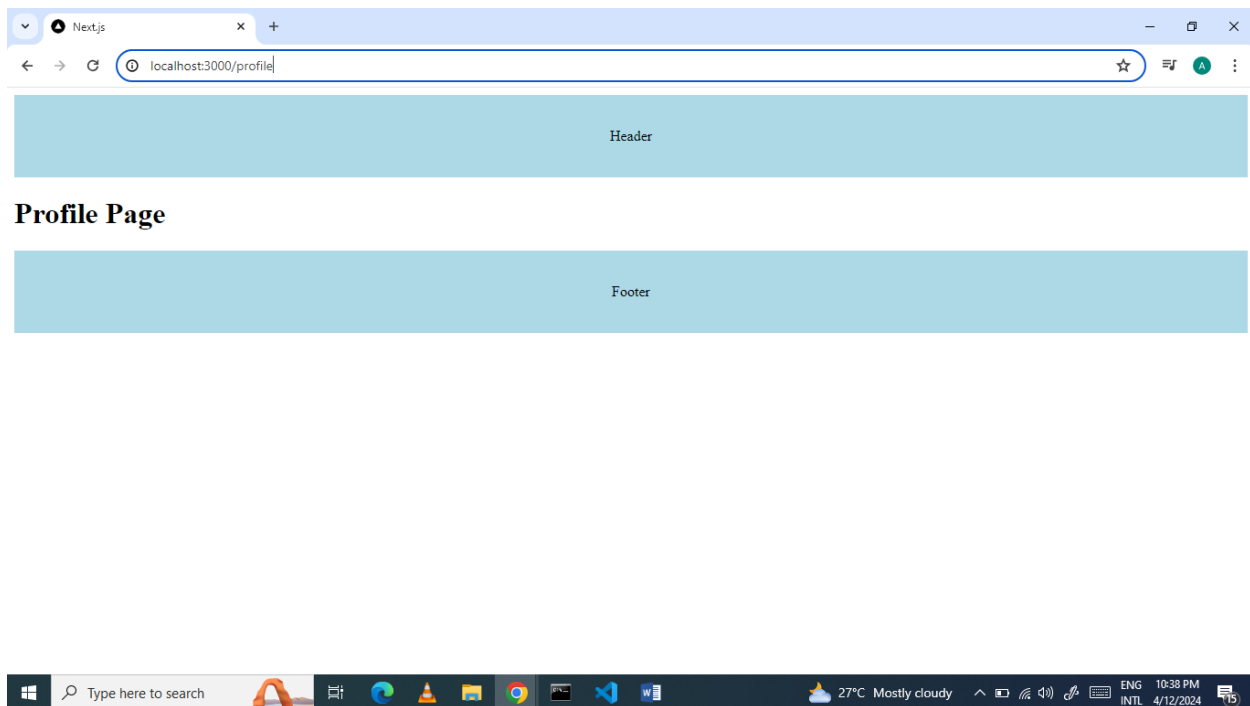
"Go to the browser and navigate to <http://localhost:3000/profile>. The output will be as shown in the screenshots below."



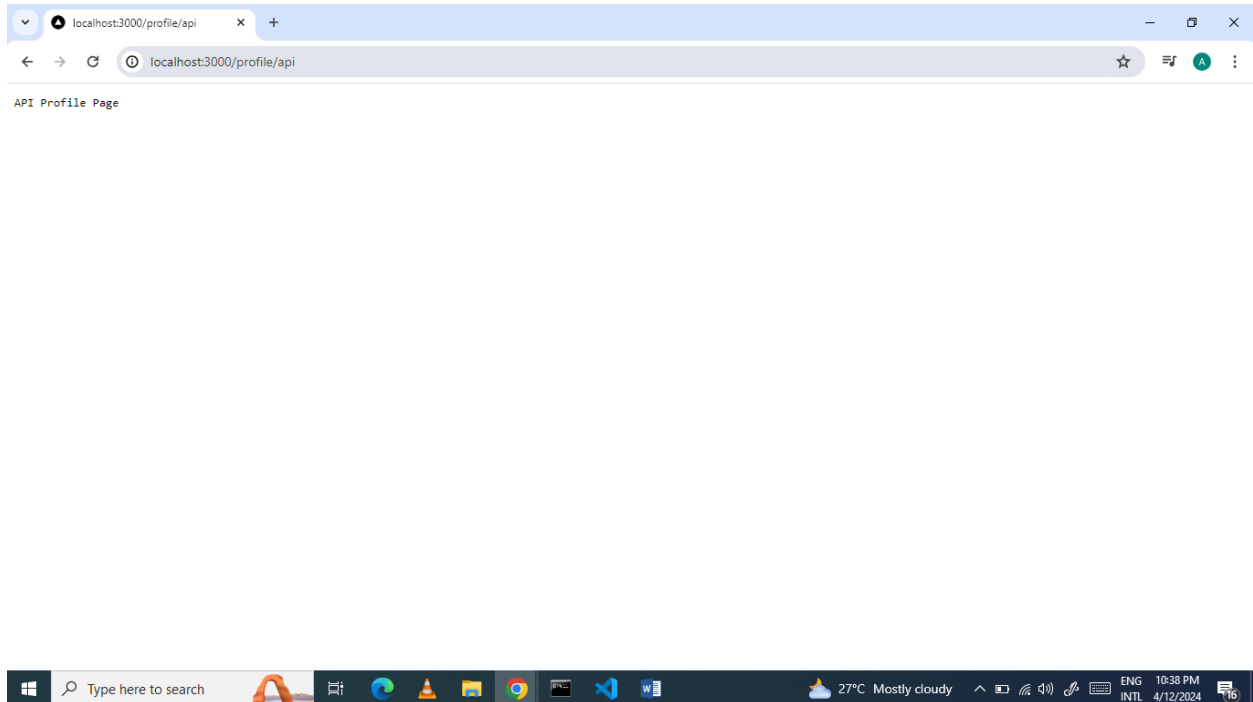
To avoid this conflict between `page.tsx` and `route.ts` and have both of them working, you can move the route handler to an "api" subdirectory



"Go to your browser and navigate to `http://localhost:3000/profile`. The output will be as shown in the screenshots below."



"Go to your browser and navigate to `http://localhost:3000/profile/api`. The output will be as shown in the screenshots below."



"Route handlers let you make your own special handlers for specific routes. You set them up in a file named `route.ts` within the 'app' folder. If you put a `route.ts` file at the same level as a `page.tsx` file, it causes a problem and the page (`page.tsx`) won't work."