

**Made By: Mahnoor (00070019) Sunday 9 to 12**

## **DAY 3 API Integration & Migration**

### **Day 3 - API Integration Report – FOOD TUCK**

#### API INTEGRATION PROCESS

##### 1. Environment Setup:

a. Install Sanity: `npm create sanity@latest ---template clean --create-project "learning-sanity-project" --dataset production`

##### b. Setting Up Environment Variables

Set up our environment variables. Create a `.env.local` file in your project root. Add the following variables:

`NEXT_PUBLIC_SANITY_PROJECT_ID=your_project_id`

`NEXT_PUBLIC_SANITY_DATASET=production`

`SANITY_API_TOKEN=your_sanity_token`

##### c. Data Fetching:

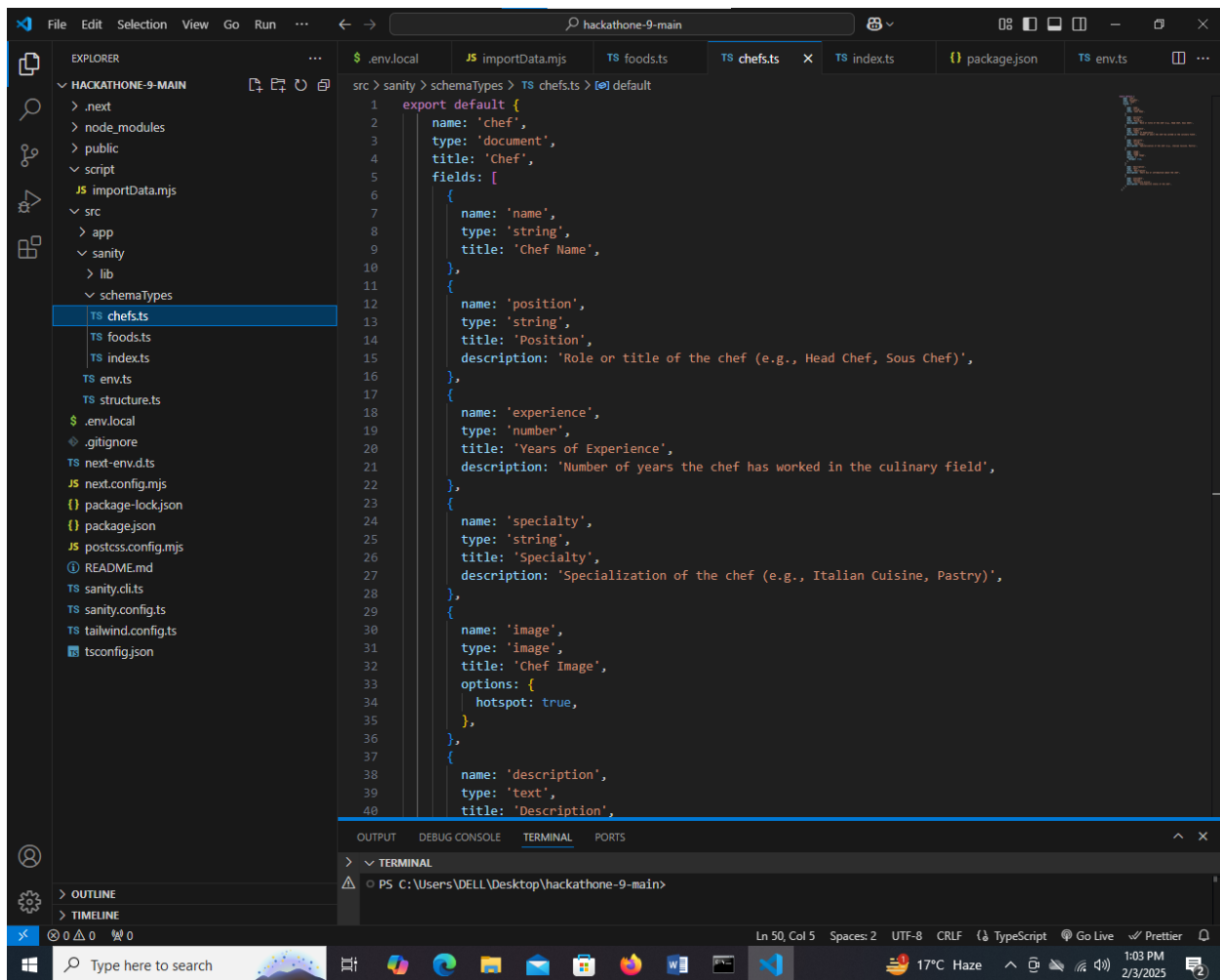
Used Axios to retrieve product data from an external API (foods and chefs).

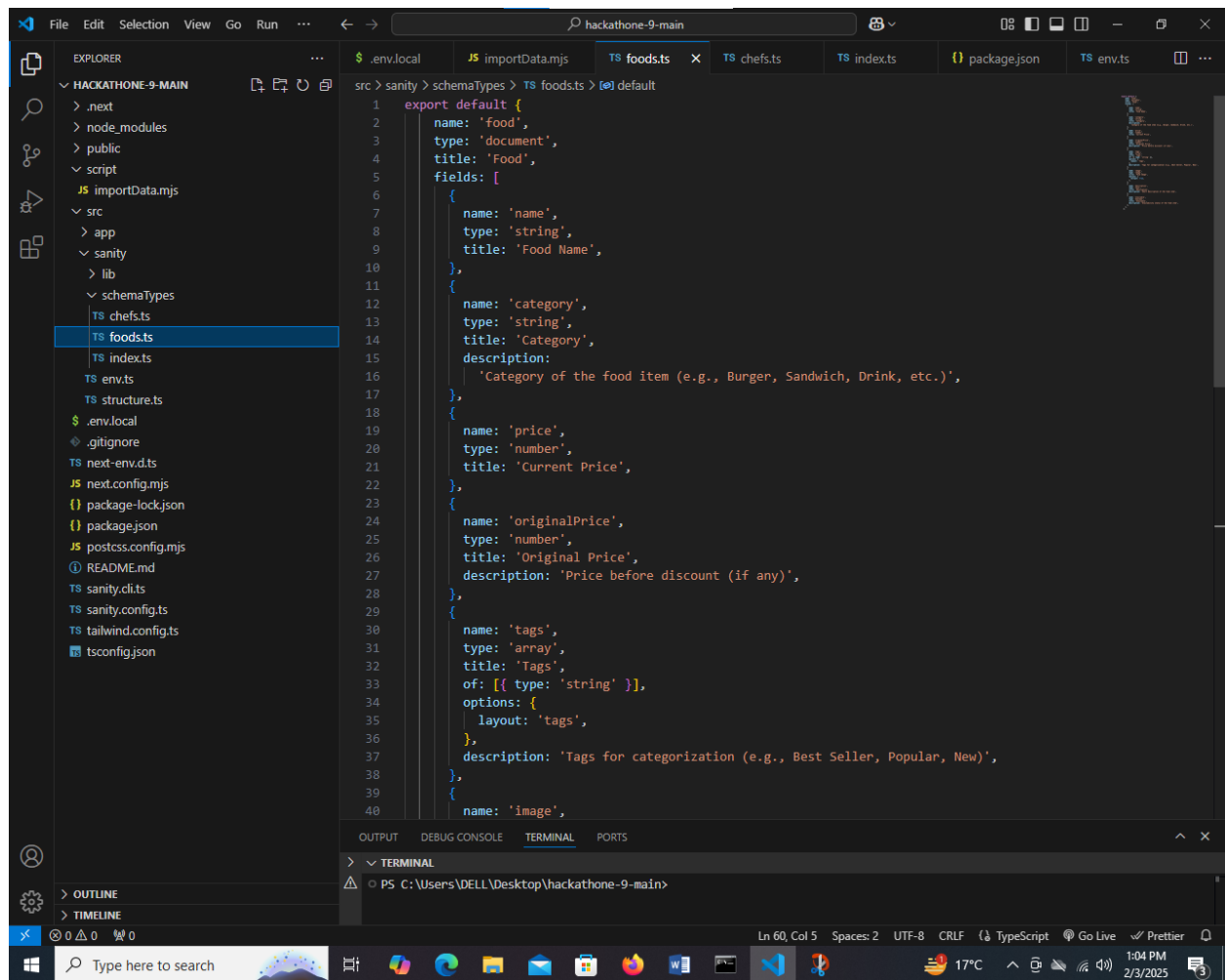
Accessed the following Endpoints:

`https://sanity-nextjs-rouge.vercel.app/api/foods`

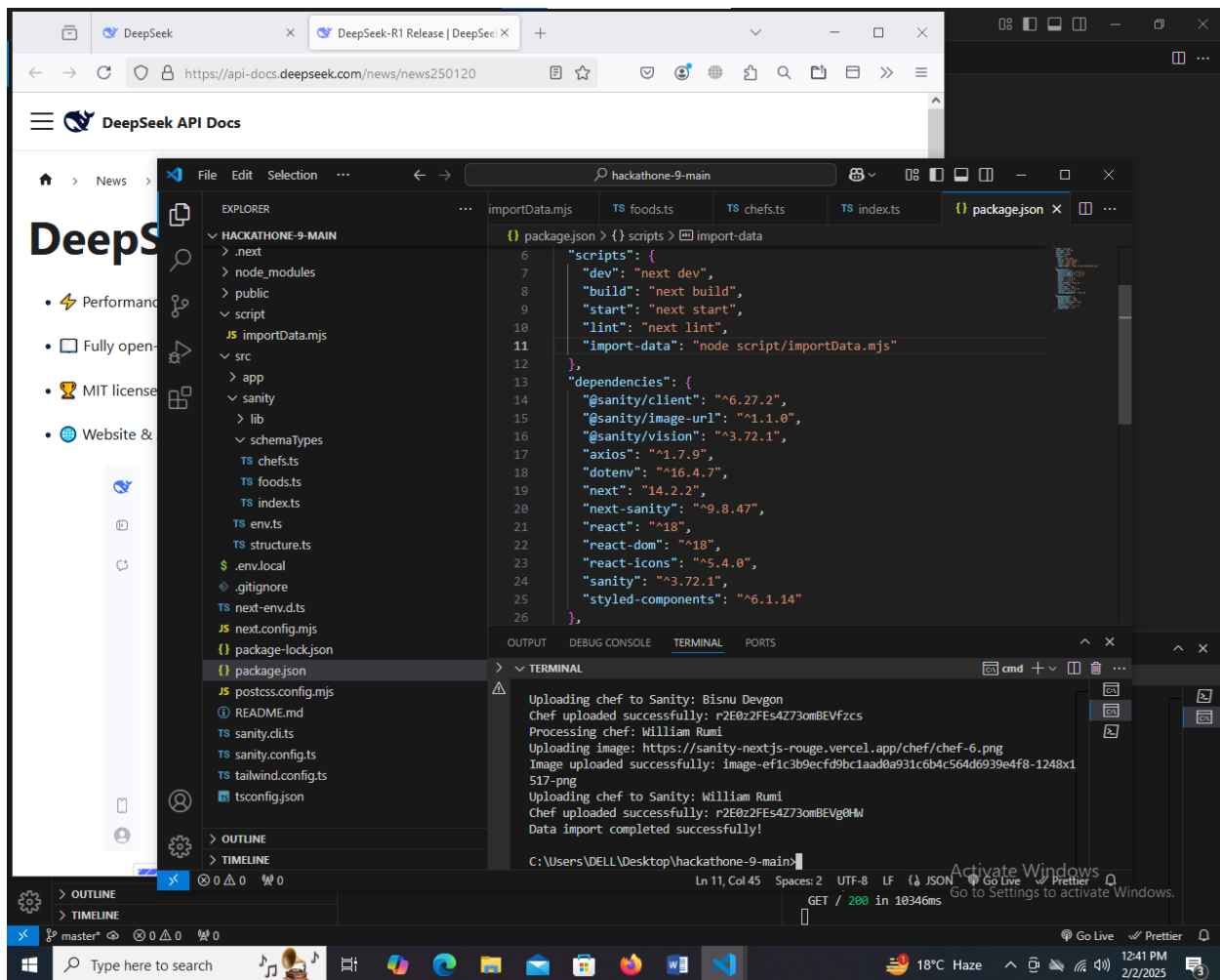
`https://sanity-nextjs-rouge.vercel.app/api/chefs`

## **2. Adjustments made to schemas**

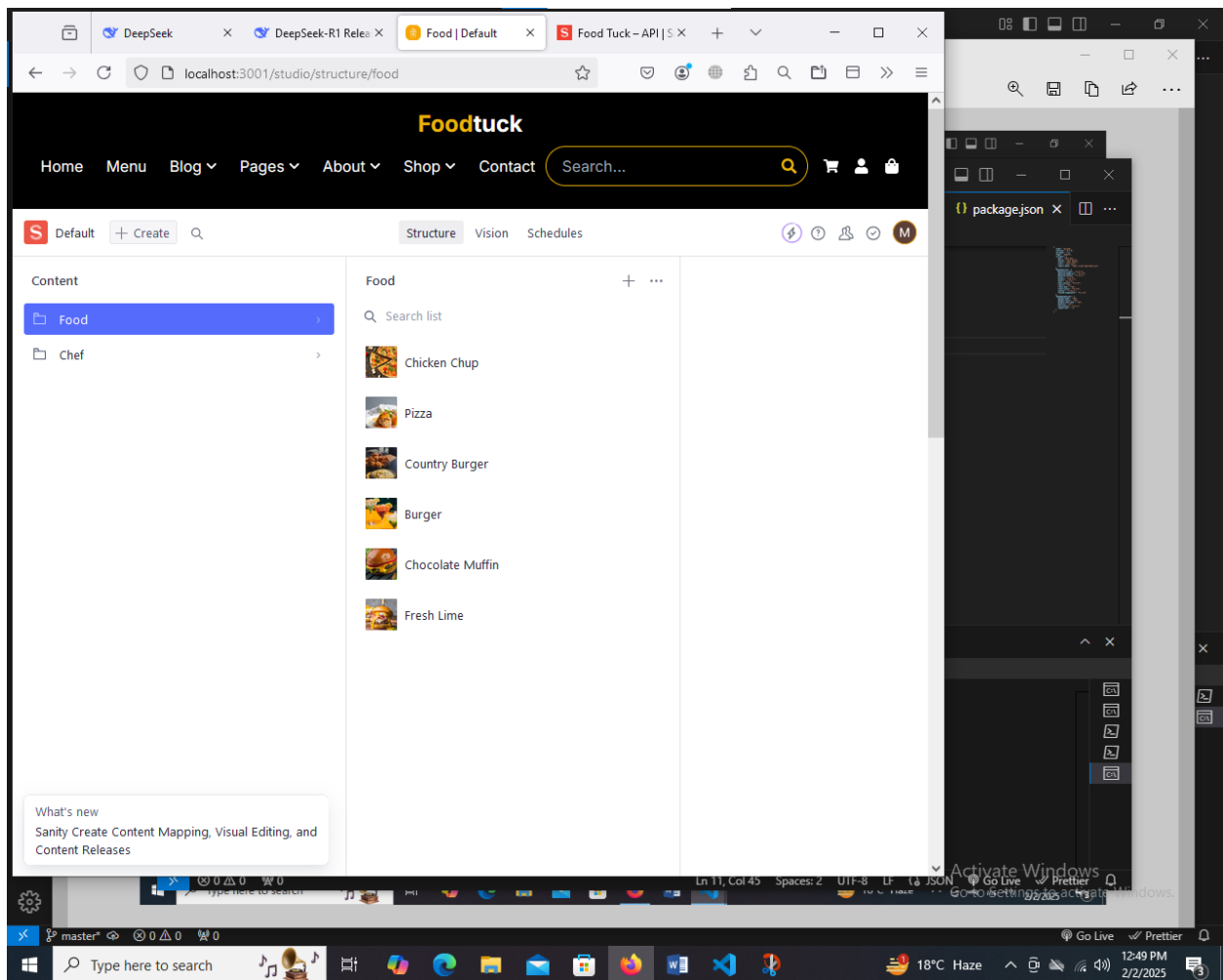




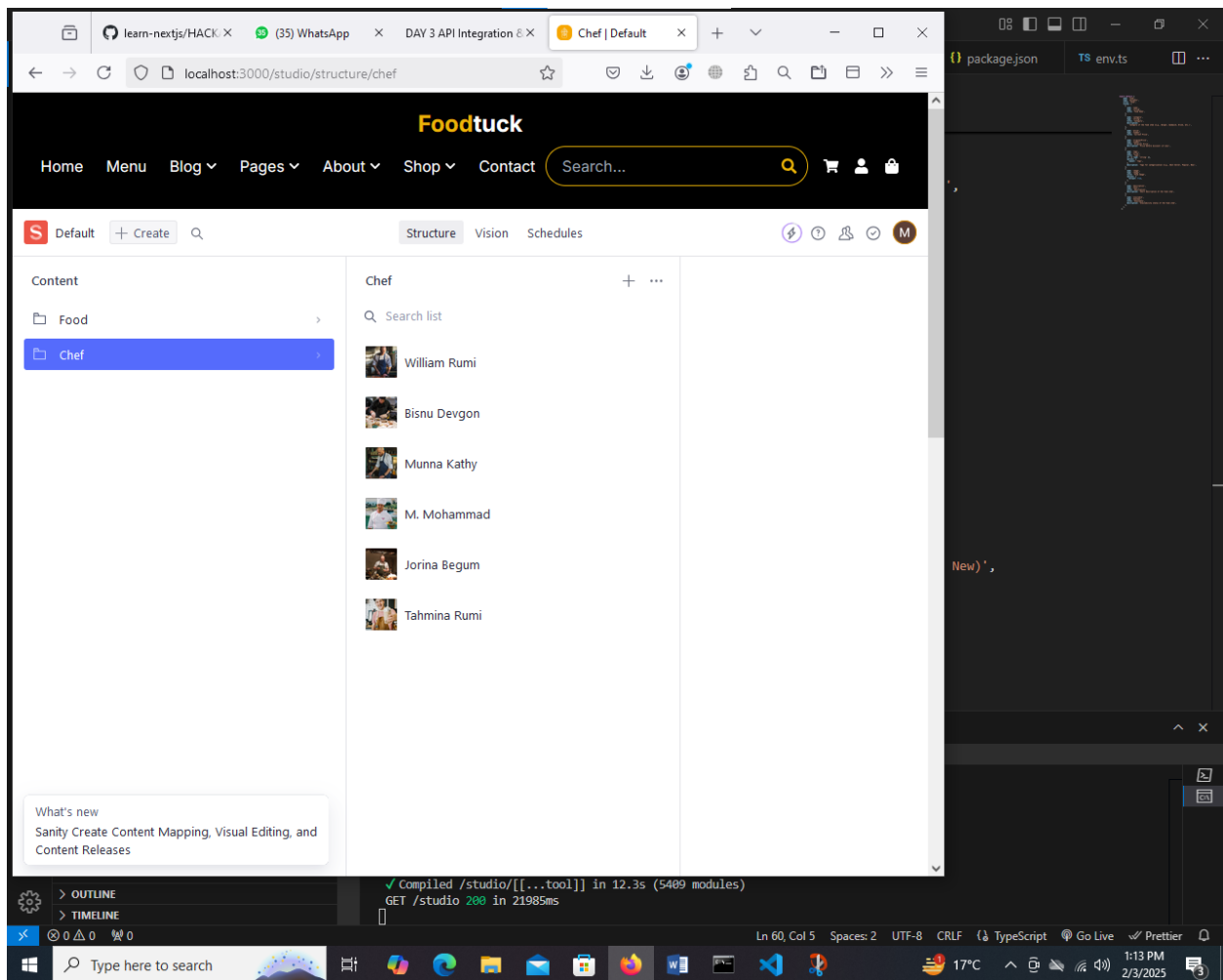
### 3. Import data to sanity:



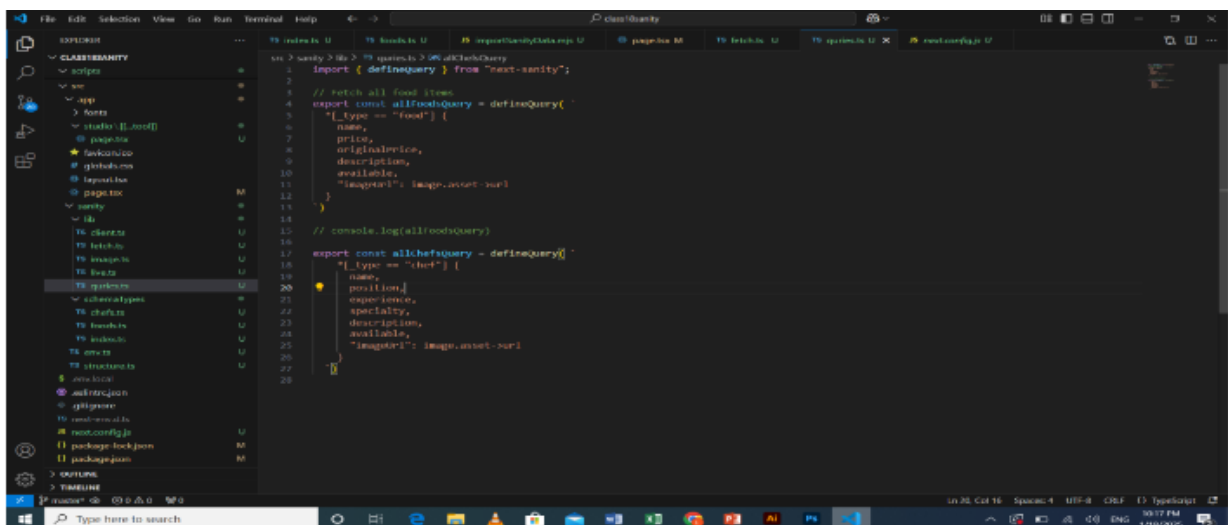
**Output: Finally show data. (Food)**

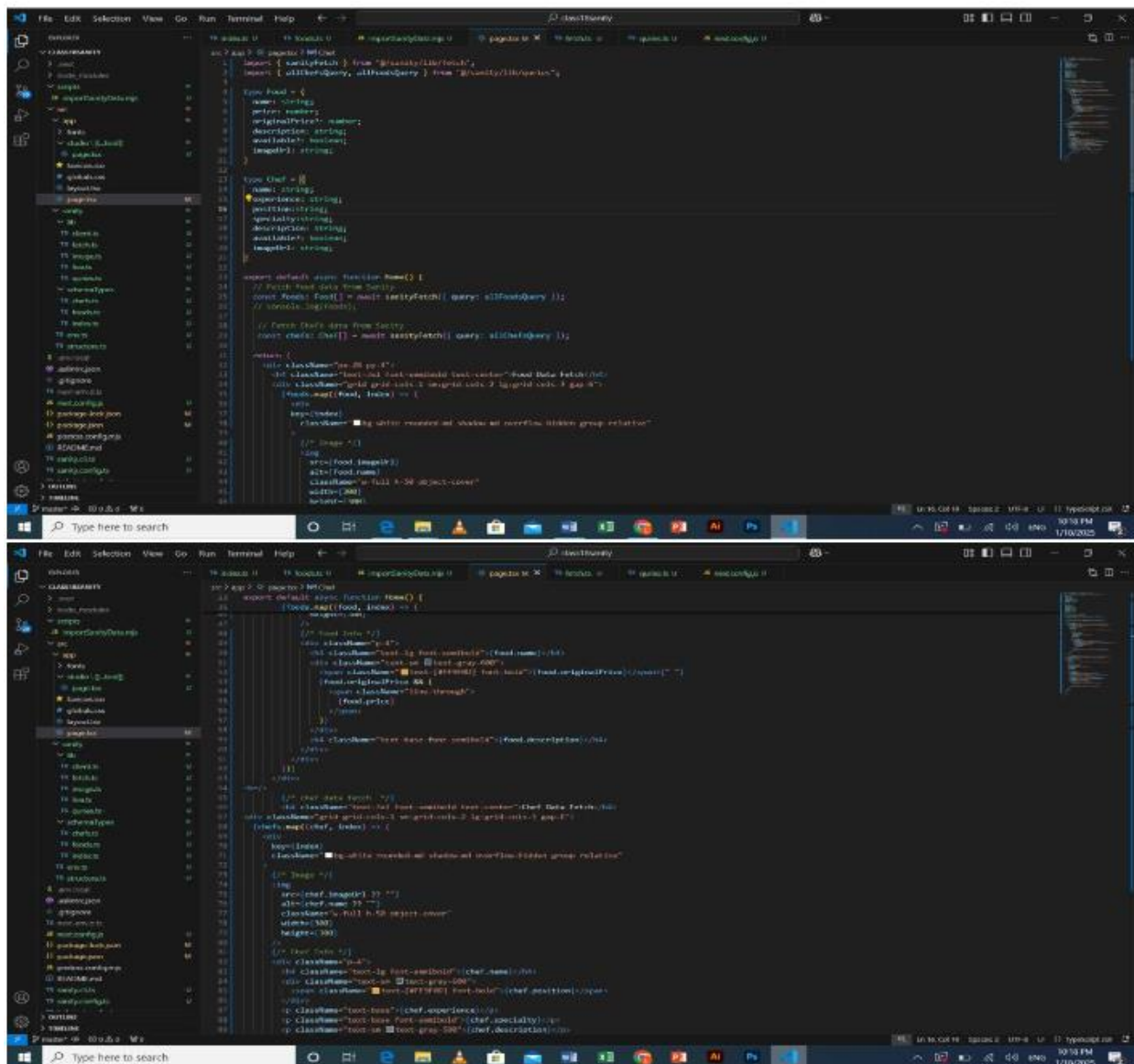


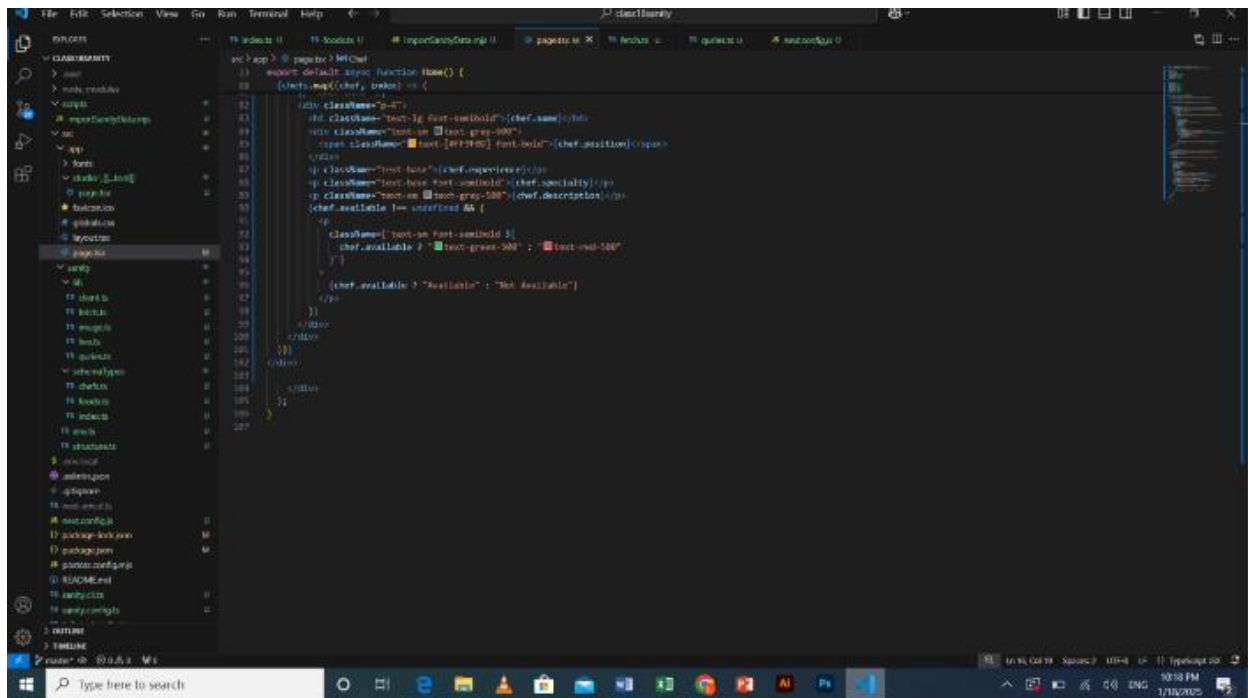
**Chef Data.**



Data successfully displayed

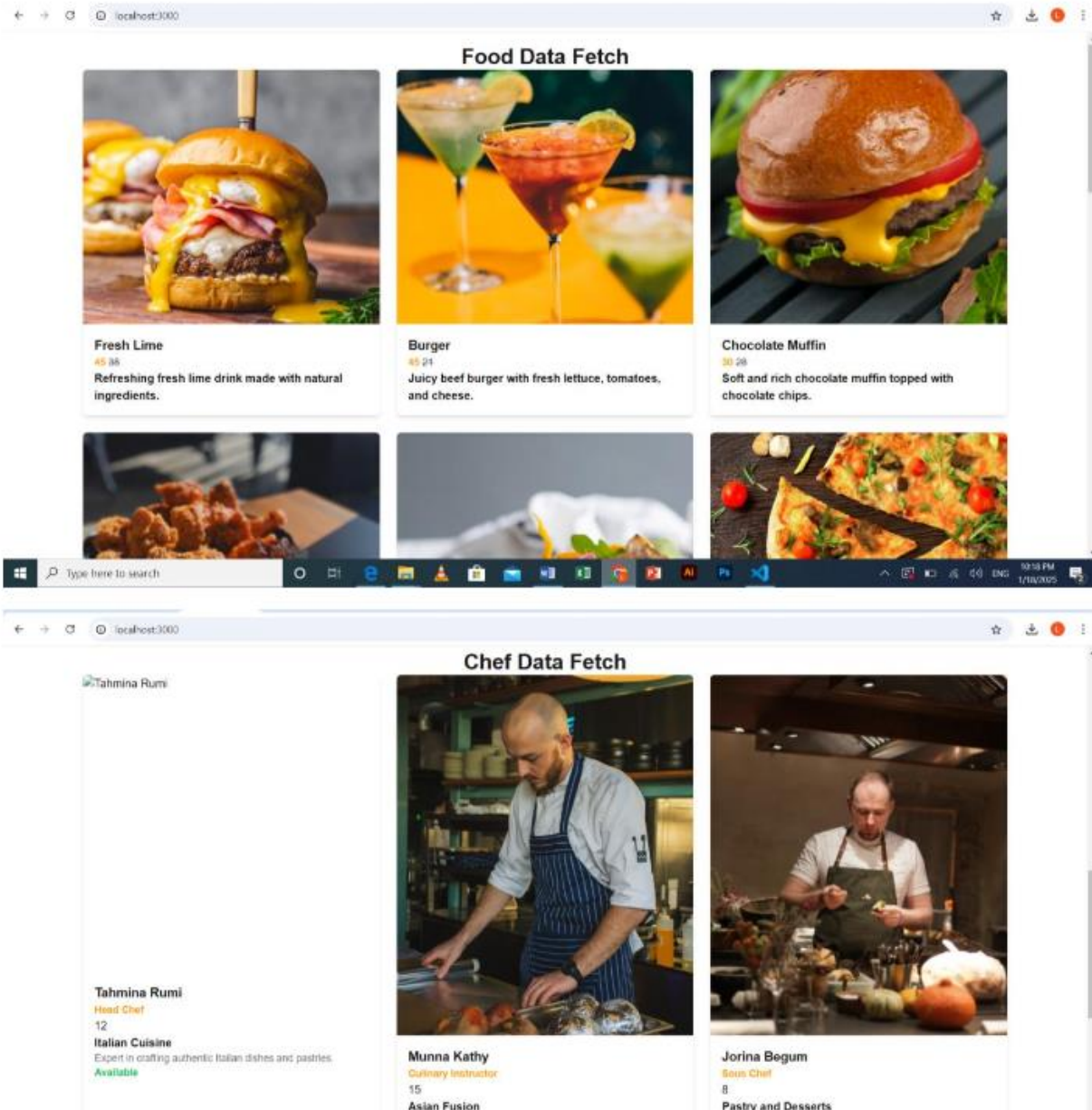






Made by: Muhammad Bilal Qureshi id:00075879 Sunday 2 to 5





## Goal of This Task:

- Establish a connection with Sanity CMS for efficient and streamlined product data management.
- Utilize GROQ queries to dynamically retrieve well-structured data from Sanity.
- Adapt and refine the Sanity schema to align with evolving API requirements.
- Migrate product data from an external API to Sanity CMS, ensuring images are uploaded successfully.

- Test and validate the integration to ensure seamless data flow between the backend and frontend.

**Made by: Muhammad Bilal Qureshi id:00075879 Sunday 2 to 5**