

HACKATHON 3 DAY 3

API INTEGRATION AND DATA MIGRATION

Document prepared by Karim Buksh

Class Sunday 2 to 5



INTRODUCTION

- The task for Day 3 of Hackathon 3 was given by Sir Ameen Alam in a document. The main goal was to integrate APIs and move data into Sanity CMS to create a working marketplace backend. This task showed how to use APIs, transfer data into Sanity CMS, and make sure it works well with my Marketplace Template 2.
- The key steps were:
 1. Understanding the given API.
 2. Checking and fixing the schema.
 3. Learning how to move data.
 4. Adding the API to Next.js.
- All steps and screenshots are included in the document for reference.

PROJECT CREATED IN SANITY.IO NAMED HACKATHON_3

The screenshot shows the Sanity.io project management interface for a project named 'Hackathon_3'. The browser address bar displays the URL `sanity.io/manage/personal/project/yvqok3u1`. The user is logged in as 'Karim KB Gabol'. The project details show a 'Growth Trial' plan, 'Active' status, and project ID 'yvqok3u1'. A navigation bar includes links for 'Getting started', 'Overview' (selected), 'Members', 'Studios', 'Datasets', 'Access', 'Activity', 'Usage', 'Plan', 'API', and 'Settings'. The 'Next steps' section provides instructions to initialize the project with the CLI, including a terminal command: `npm create sanity@latest -- --project yvqok3u1 --dataset produ`. The 'Project members' section shows an invitation to add the first team member with a button labeled '+ Invite project members'. The Windows taskbar at the bottom shows the time as 11:19 PM on 1/18/2025.

sanity.io/manage/personal/project/yvqok3u1

Karim KB Gabol Hackathon_3 30 days left in trial

Karim KB Gabol

Hackathon_3

PLAN: Growth Trial STATUS: Active PROJECT ID: yvqok3u1

Getting started Overview Members Studios Datasets Access Activity Usage Plan API Settings

Next steps

Initialize your project with the CLI

Run this command in your Terminal to continue setting up your project.

```
npm create sanity@latest -- --project yvqok3u1 --dataset produ
```

Copy

Having issues with the CLI?

What's new

Sanity Create Content Mapping, Visual Editing, and Content Releases

View more →

Project members View all →

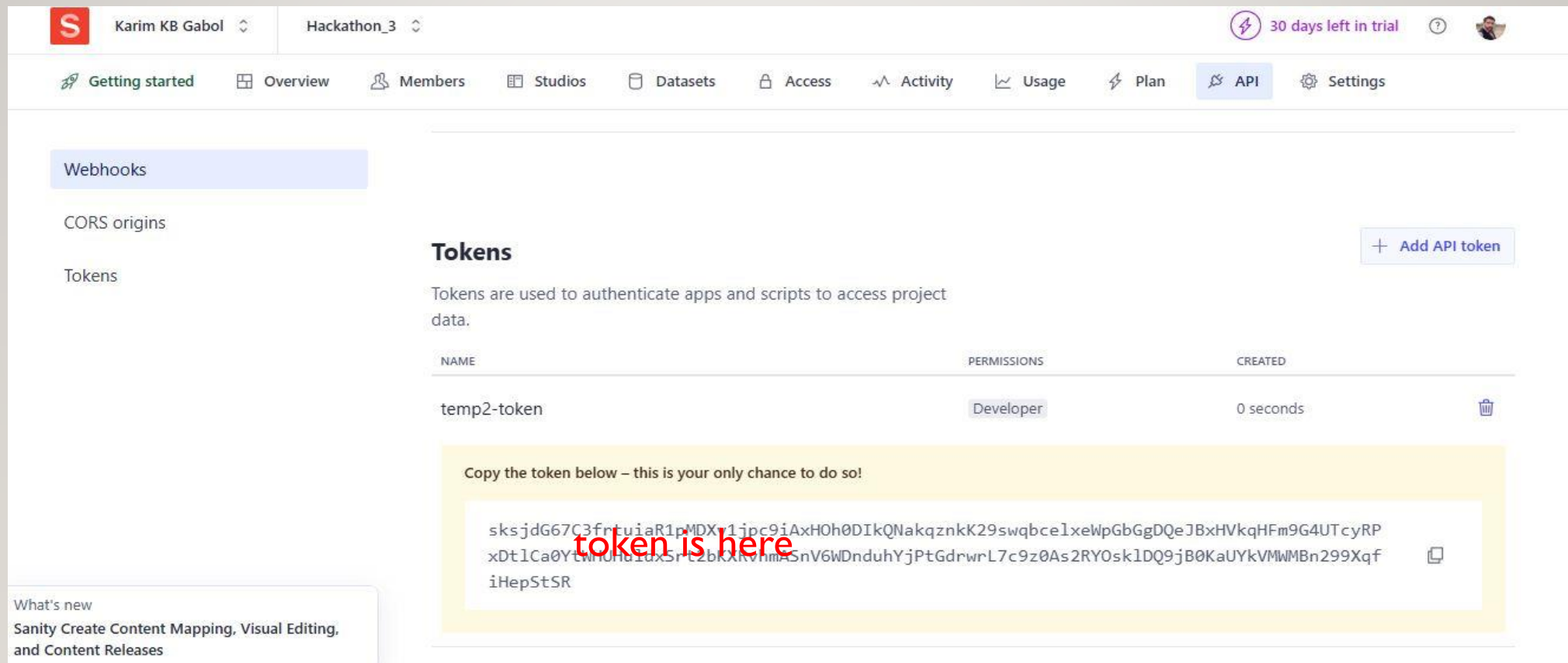
Invite your first team member

+ Invite project members

Type here to search

11:19 PM 1/18/2025

TOKEN GENERATED FOR MY PROJECT IN SANITY/PROJECT/API → TOKENS



The screenshot shows the Sanity dashboard for a project named 'Hackathon_3'. The user is 'Karim KB Gabol'. The 'API' tab is selected in the top navigation bar. On the left sidebar, 'Webhooks', 'CORS origins', and 'Tokens' are listed. The 'Tokens' section is active, showing a table with one token named 'temp2-token' with 'Developer' permissions, created '0 seconds' ago. A yellow callout box contains the token string and a red text overlay 'token is here' pointing to it. A 'What's new' sidebar on the bottom left mentions 'Sanity Create Content Mapping, Visual Editing, and Content Releases'.

Tokens

Tokens are used to authenticate apps and scripts to access project data.

[+ Add API token](#)

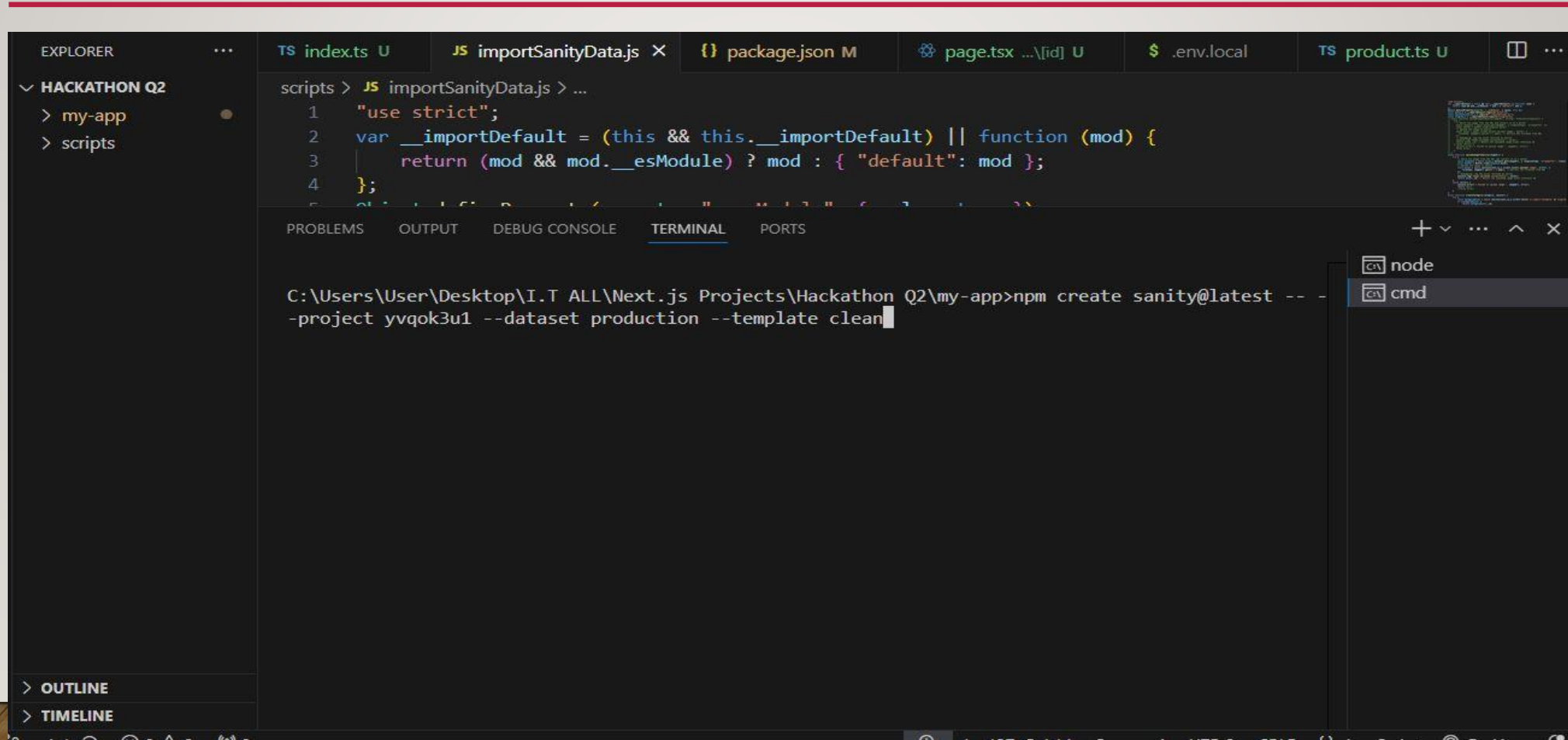
NAME	PERMISSIONS	CREATED
temp2-token	Developer	0 seconds

Copy the token below – this is your only chance to do so!

```
sksjdG67C3frtuiaR1pMDXv1ipc9iAxH0h0DIkQnakqznkK29swqbcelxeWpGbGgDQeJBxHVkqHFm9G4UTcyRPxDt1Ca0YtWnUhd1uxSfT2bKXkvMmASnV6WdnduhYjPtGdrwrL7c9z0As2RY0sk1DQ9jB0KaUYkVMWMBn299XqfiHepStSR
```

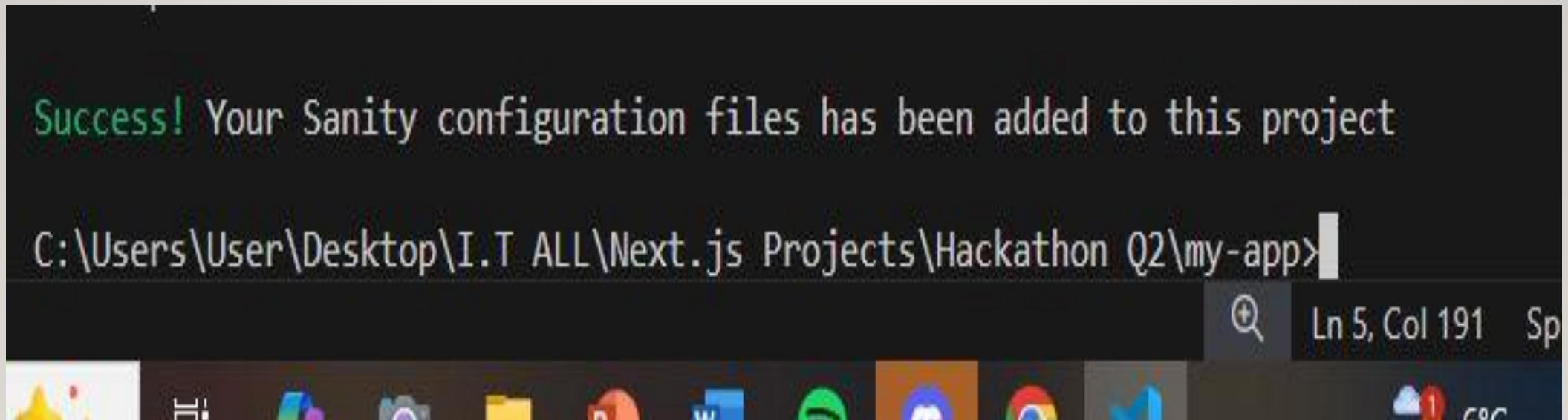
What's new
Sanity Create Content Mapping, Visual Editing, and Content Releases

SANITY INSTALL BY GIVING COMMAND IN CMD.

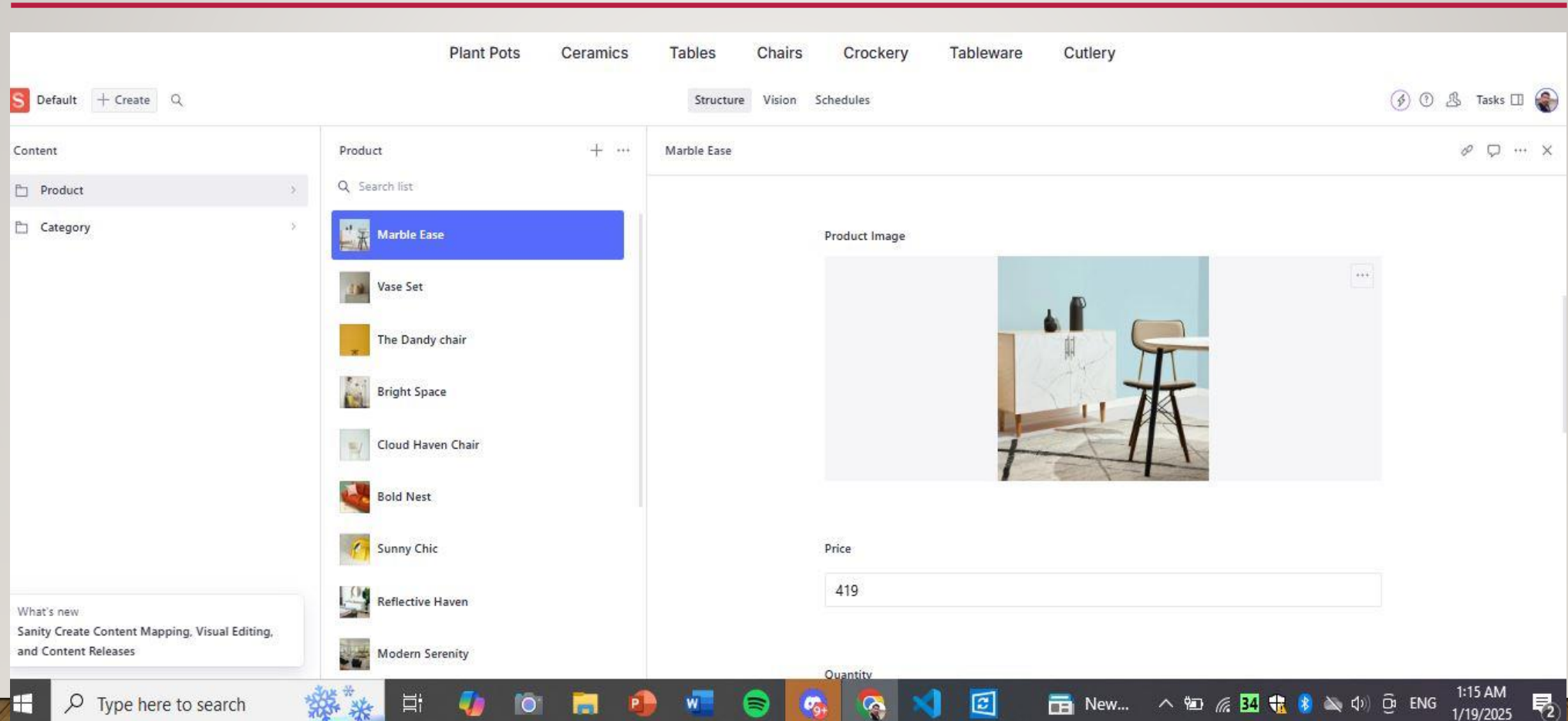


SANITY INSTALLED SUCCESSFULLY IN MY NEXT.JS PROJECT BY GIVING COMMAND IN CMD.

```
Success! Your Sanity configuration files has been added to this project  
C:\Users\User\Desktop\I.T ALL\Next.js Projects\Hackathon Q2\my-app>
```

A screenshot of a Windows Command Prompt window. The background is black with white text. The first line of text is "Success! Your Sanity configuration files has been added to this project", where "Success!" is in green. The second line shows the current directory path: "C:\Users\User\Desktop\I.T ALL\Next.js Projects\Hackathon Q2\my-app>". The window has a standard Windows taskbar at the bottom with various application icons and a system tray on the right showing a search icon, "Ln 5, Col 191", "Sp", and a weather icon with "6°C".

DATA MIGRATED INTO SANITY SUCCESSFULLY.



DEFINE QUERY IN SANITY/VISION

Plant Pots

Ceramics

Tables

Chairs

Crockery

Tableware

Cutlery

S

Default





+ Create


Q

Structure

Vision

Schedules

   Tasks 



DATASET

API VERSION

CUSTOM API VERSION

PERSPECTIVE ?


QUERY URL [COPY TO CLIPBOARD]

production

Other

v2025-01-18

raw

https://yvqok3u1.api.sanity.io/v2025-01-18/data/query/production?query: 

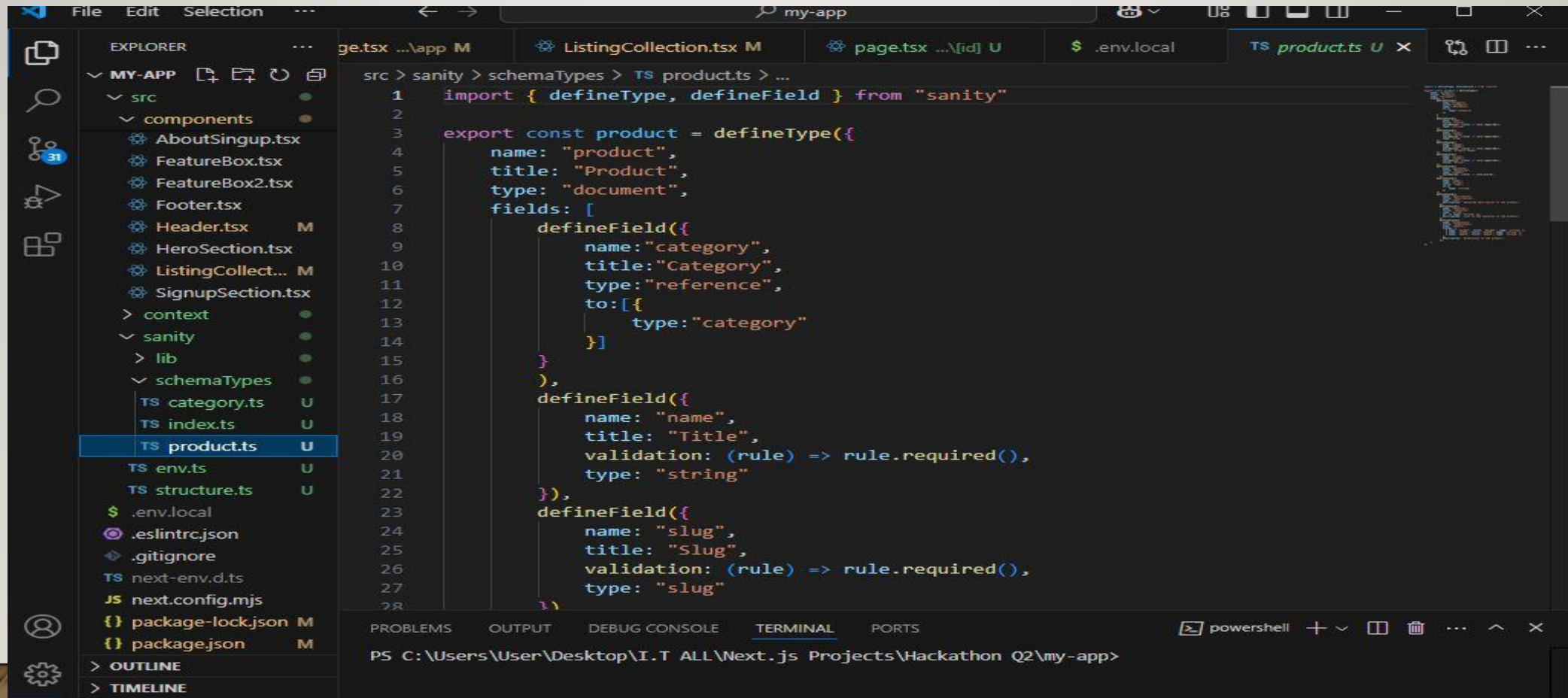
QUERY

RESULT

```
*[_type=="product"]{
  _id,
  name,
  description,
  price,
  category,
  "image_url":image.asset->url,
}
```

```
[...] 4 items
  0: {...} 6 properties
    description: A timeless design, with premium materials features as one of our most popular and iconic pieces. The dandy chair is perfect for any stylish living space with beech legs and lambskin leather upholstery.
    price: 980
    category: {...} 2 properties
      _type: reference
      _ref: tableware-1
      image_url: https://cdn.sanity.io/images/yvqok3u1/production/9b6a4fc8c65bbb4e5793fb0e1116b510d73dc9e8-630x375.png
      _id: product-1
      name: The Poplar suede sofa
  1: {...} 6 properties
    image url: https://cdn.sanity.io/images/yvqok3u1/production/f575d30bda78fa7680a74e131a9b25add2247210-5501x3095.jpg
```


SCHEMA CREATED IN SCHEMA-TYPES

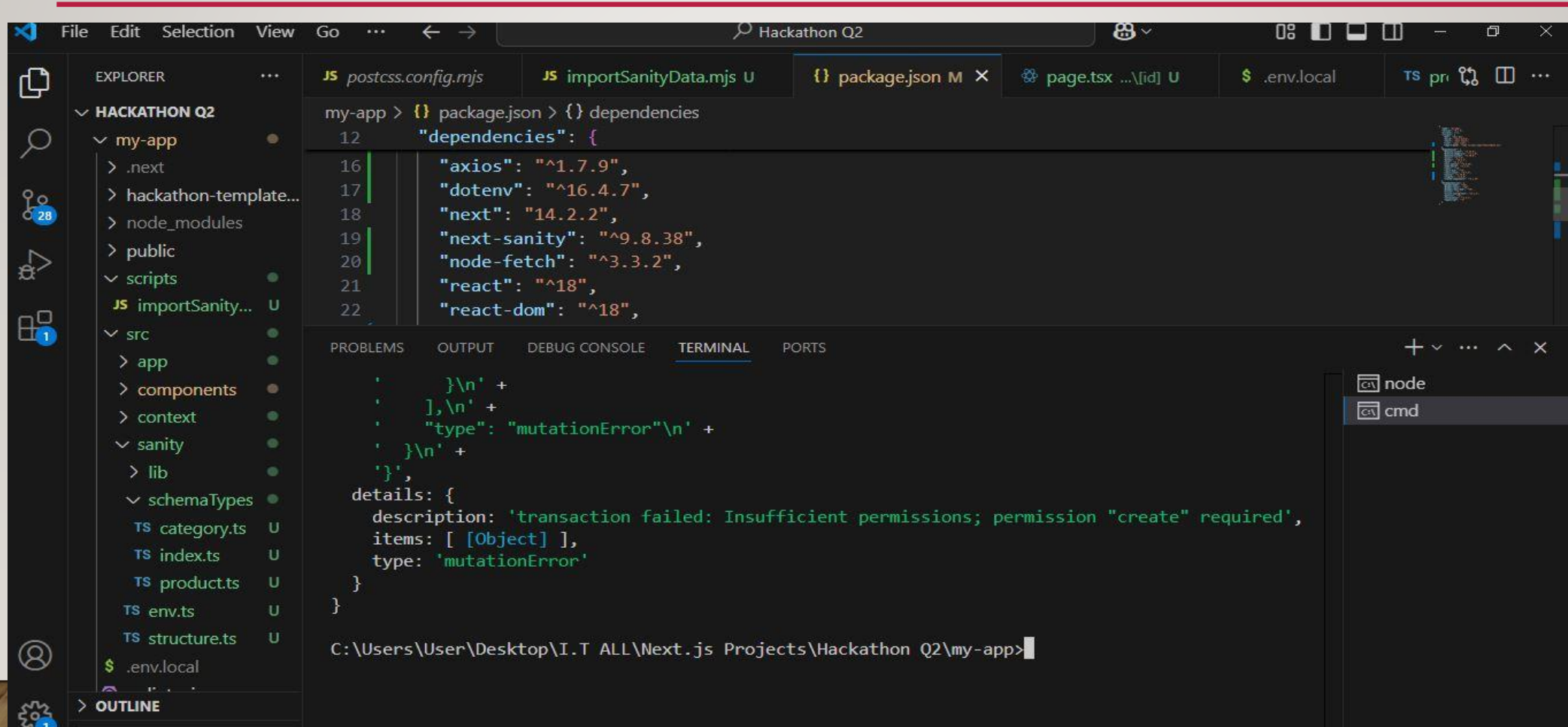


The screenshot shows a Visual Studio Code editor with a project named 'my-app'. The Explorer sidebar on the left shows the file structure, with 'src > sanity > schemaTypes > TS product.ts' selected. The main editor displays the content of 'TS product.ts', which defines a Sanity schema for a product. The code uses the 'defineType' and 'defineField' functions from the 'sanity' package. The schema includes a 'category' field of type 'reference' and 'name' and 'slug' fields of type 'string', both of which are required.

```
src > sanity > schemaTypes > TS product.ts > ...
1  import { defineType, defineField } from "sanity"
2
3  export const product = defineType({
4    name: "product",
5    title: "Product",
6    type: "document",
7    fields: [
8      defineField({
9        name: "category",
10       title: "Category",
11       type: "reference",
12       to: [{
13         type: "category"
14       }]
15     }),
16     defineField({
17       name: "name",
18       title: "Title",
19       validation: (rule) => rule.required(),
20       type: "string"
21     }),
22     defineField({
23       name: "slug",
24       title: "Slug",
25       validation: (rule) => rule.required(),
26       type: "slug"
27     })
28   ]
29 })
```

At the bottom of the editor, the TERMINAL panel shows the current directory path: `PS C:\Users\User\Desktop\I.T ALL\Next.js Projects\Hackathon Q2\my-app>`.

DATA IS IMPORTING FROM SANITY.



DATA IMPORTED COMPLETED SUCCESSFULLY..

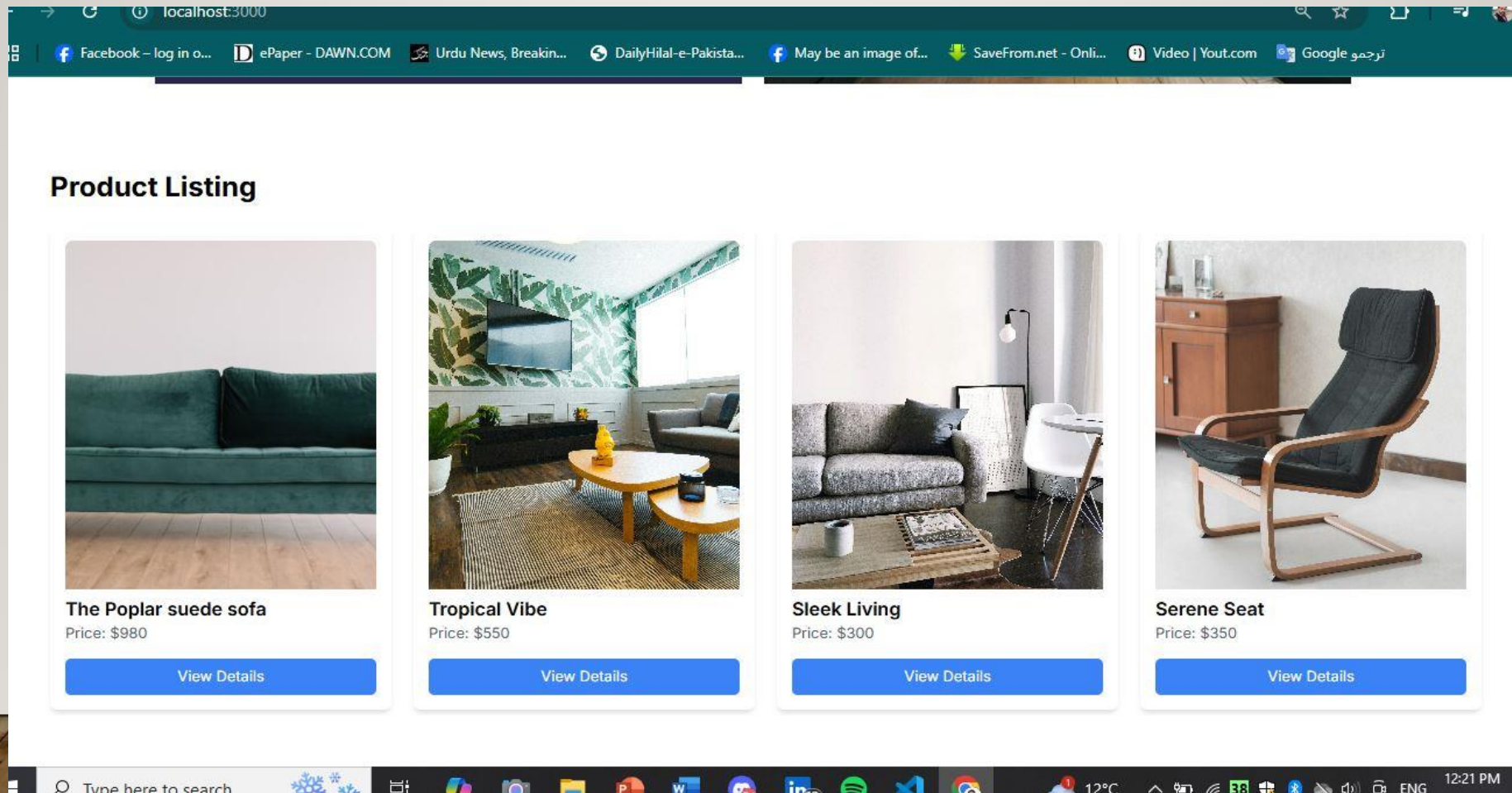
The screenshot displays the Visual Studio Code interface for a project named 'Hackathon Q2'. The Explorer sidebar on the left shows the project structure, including a 'src' directory with 'app', 'components', 'context', and 'sanity' subdirectories, and a 'lib' directory containing TypeScript files like 'client.ts', 'image.ts', 'live.ts', 'category.ts', 'index.ts', 'products.ts', 'env.ts', and 'structure.ts'. The main editor area shows the 'image.ts' file with a long string of alphanumeric characters. The Terminal panel at the bottom displays the output of a command, showing the successful upload of an image and the creation of a category named 'Crockory'. The status bar at the bottom indicates the current file is 'image.ts' at line 4, column 191.

```
my-app > $ .env.local
1
2
3 laTMYFunVcJC6pyGasgva6xhDaR2gSdiMTv84WdMib0aH1Tj7EjzG8gZD6rAwF4Aj4VXSJN3AwZ9vDvu7F0itQG36qBQgIkksI
4 'RPxDt1Ca0YtWHUHuldxSrt2bKXRvhmASnV6WDnduhYjPtGdrwrL7c9z0As2RY0sk1DQ9jB0KaUYkVMWMBn299XqfiHepStSR"
```

```
Processing product: Marble Ease
Uploading image: https://cdn.sanity.io/images/ri847jqu/production/7f6146b6625bf94030f2aa896e9b0c847e5f4919-8000x8000.jpg
Image uploaded successfully: image-7f6146b6625bf94030f2aa896e9b0c847e5f4919-8000x8000-jpg
Category created successfully {
  _createdAt: '2025-01-18T20:11:46Z',
  _id: 'crockory-24',
  _rev: 'YjFIo1g1LQZHJZg27Q4V4Z',
  _type: 'category',
  _updatedAt: '2025-01-18T20:11:46Z',
  name: 'Crockory',
  slug: { _type: 'slug', current: 'crockory' }
}
✓ Imported product: Marble Ease
✓ Data import completed!
```

C:\Users\User\Desktop\I.T ALL\Next.js Projects\Hackathon Q2\my-app>

DATA FETCHED INTO MY NEXT.JS PROJECT AND RENDERED SUCCESSFULLY.. CONGRATULATIONS...



CONCLUSION

- On Day 3 of the Hackathon, I focused on integrating APIs and migrating data into Sanity CMS to create a functional marketplace backend. This day turned out to be incredibly productive and rewarding. Although the task initially appeared daunting, tackling and resolving numerous errors along the way gave me a deep sense of accomplishment. The challenges not only tested my skills but also enhanced my confidence. Completing the task successfully filled me with immense satisfaction and joy. I am truly enjoying the learning journey each day brings during this hackathon.

DAY 3 CHECKLIST

- **Self-Validation Checklist:**
- **API Understanding: ✓**
- **Schema Validation: ✓**
- **Data Migration: ✓**
- **API Integration in Next.js: ✓**
- **Submission Preparation: ✓**