API Integration Report

Marketplace: Comforty.

API integration Process.

According to my roll number I was assigned template 8. And to practice API integration and sanity I was provided a mock API to practice API integration and migrating data to sanity.

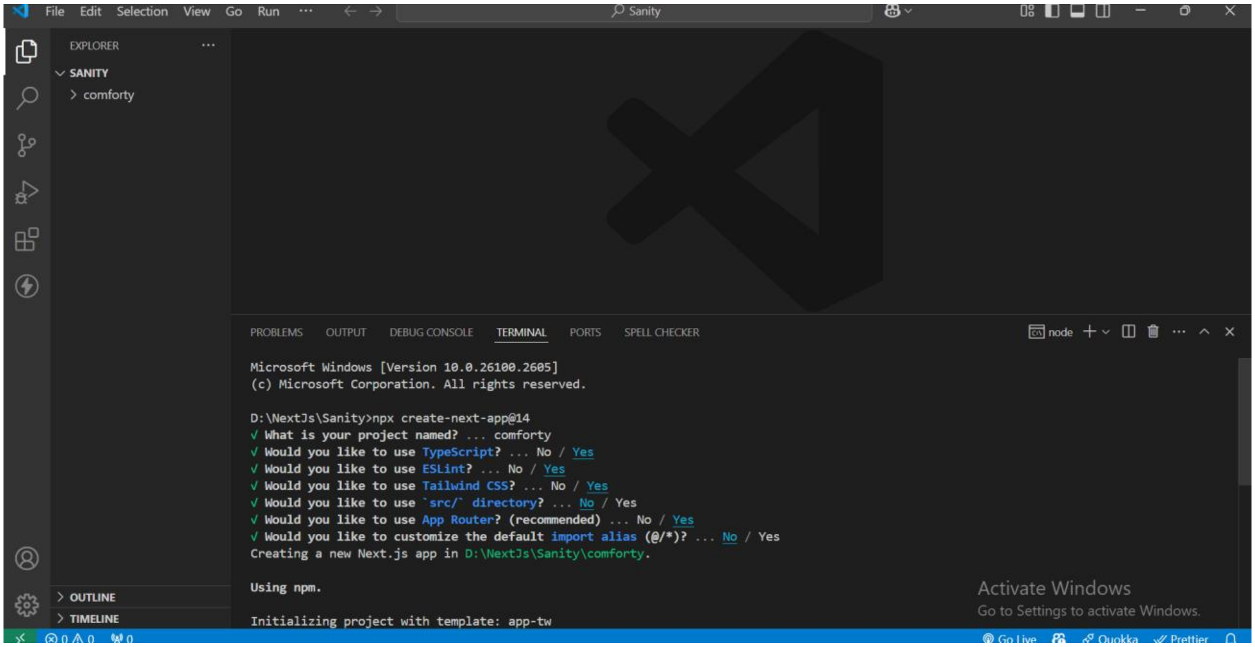
API LINK: https://giaic-hackathon-template-08.vercel.app/api/

For product: https://giaic-hackathon-template-08.vercel.app/api/products

For categories: https://giaic-hackathon-template-08.vercel.app/api/categories

Step-1:

We created our next-js project and setup our sanity studio in it.

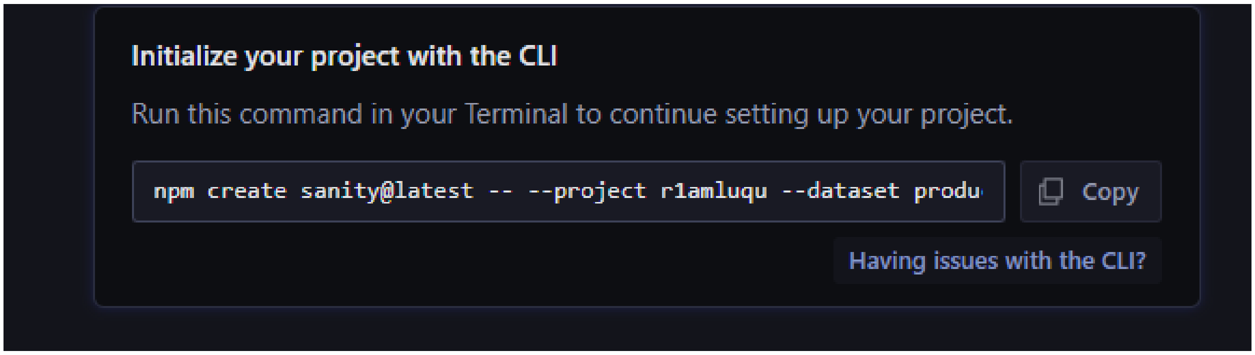


File Edit Selection View Go Run ···
Sanity
EXPLORER
SANITY
comforty
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SPELL CHECKER
node
Activate Windows
Go to Settings to activate Windows.
Go Live
Quokka
Prettier

```
Microsoft Windows [Version 10.0.26100.2605]
(c) Microsoft Corporation. All rights reserved.

D:\NextJs\Sanity>npx create-next-app@14
√ What is your project named? ... comforty
√ 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 D:\NextJs\Sanity\comforty.

Using npm.

Initializing project with template: app-tw
```

Step-2:

To use sanity in our project we used this command:



Initialize your project with the CLI

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

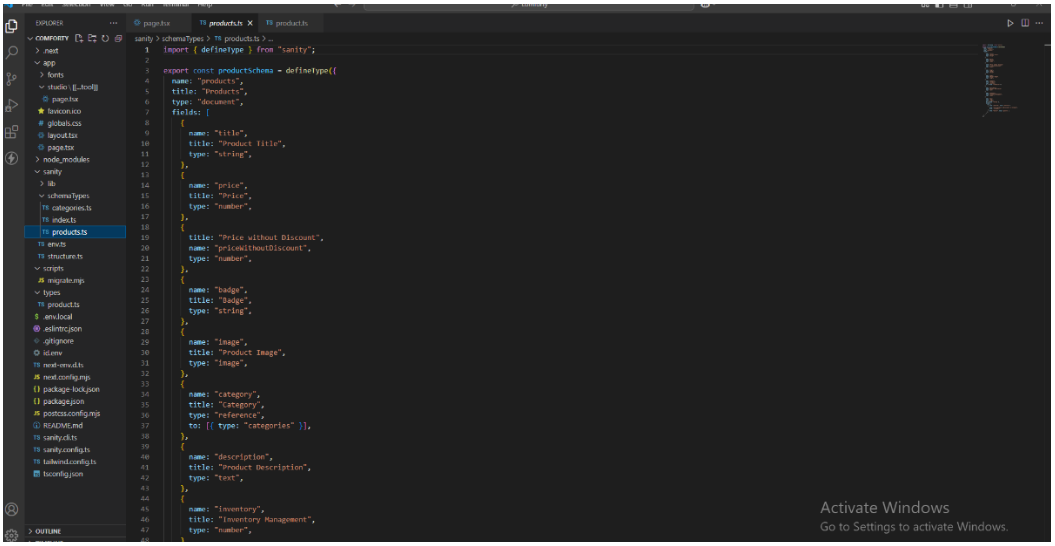
```
npm create sanity@latest -- --project r1amluqu --dataset produ
```

It allowed us to include sanity in our project and include sanity studio on our web to handle data.

Step-3:

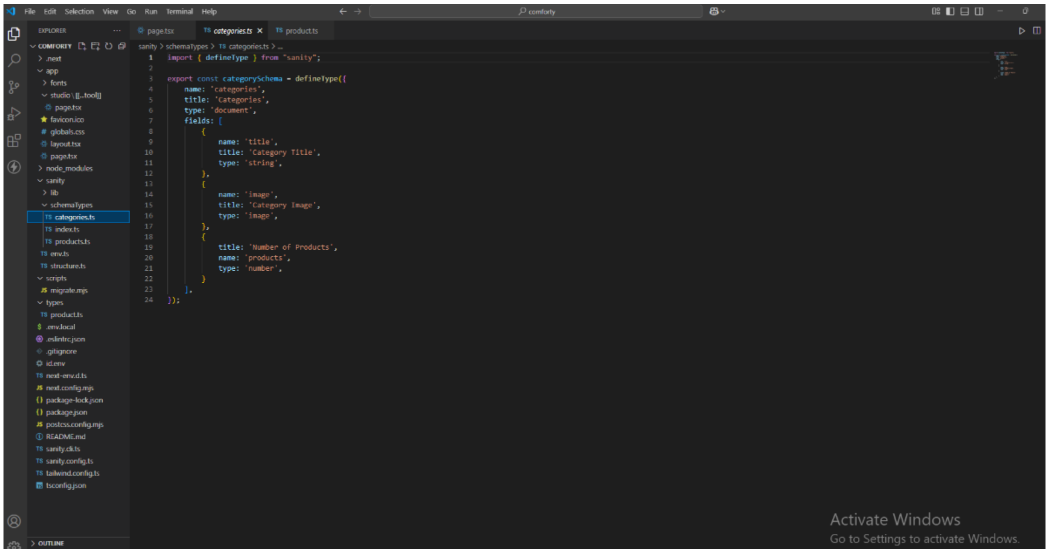
We set up our sanity and its schemas:

For this project we needed two schemas so we created two file in sanity schemaTypes folder one for products which is:



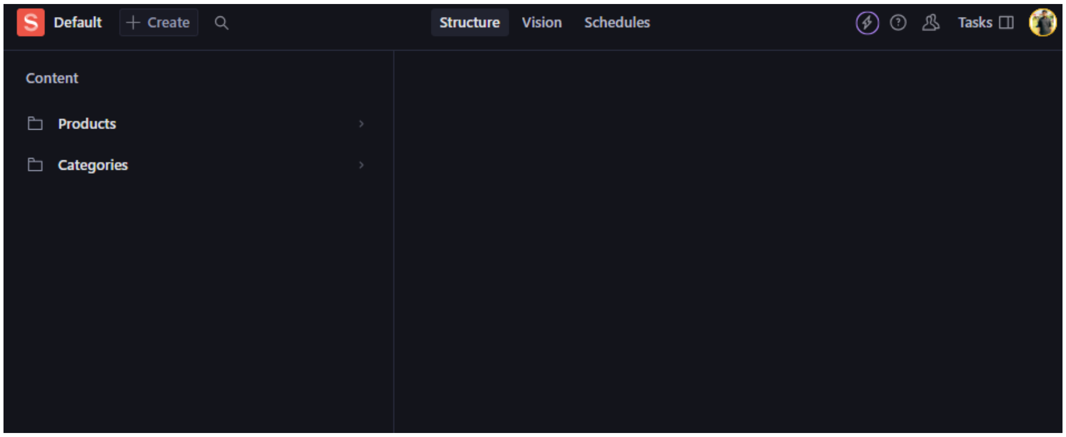
```
import { defineType } from "sanity";

export const productSchema = defineType({
  name: "products",
  title: "Products",
  type: "document",
  fields: [
    {
      name: "title",
      title: "Product Title",
      type: "string",
    },
    {
      name: "price",
      title: "Price",
      type: "number",
    },
    {
      title: "Price without Discount",
      name: "priceWithoutDiscount",
      type: "number",
    },
    {
      name: "badge",
      title: "Badge",
      type: "string",
    },
    {
      name: "image",
      title: "Product Image",
      type: "image",
    },
    {
      name: "category",
      title: "Category",
      type: "reference",
      to: [{ type: "categories" }],
    },
    {
      name: "description",
      title: "Product Description",
      type: "text",
    },
    {
      name: "inventory",
      title: "Inventory Management",
      type: "number",
```

And then set up our categories schema which is:



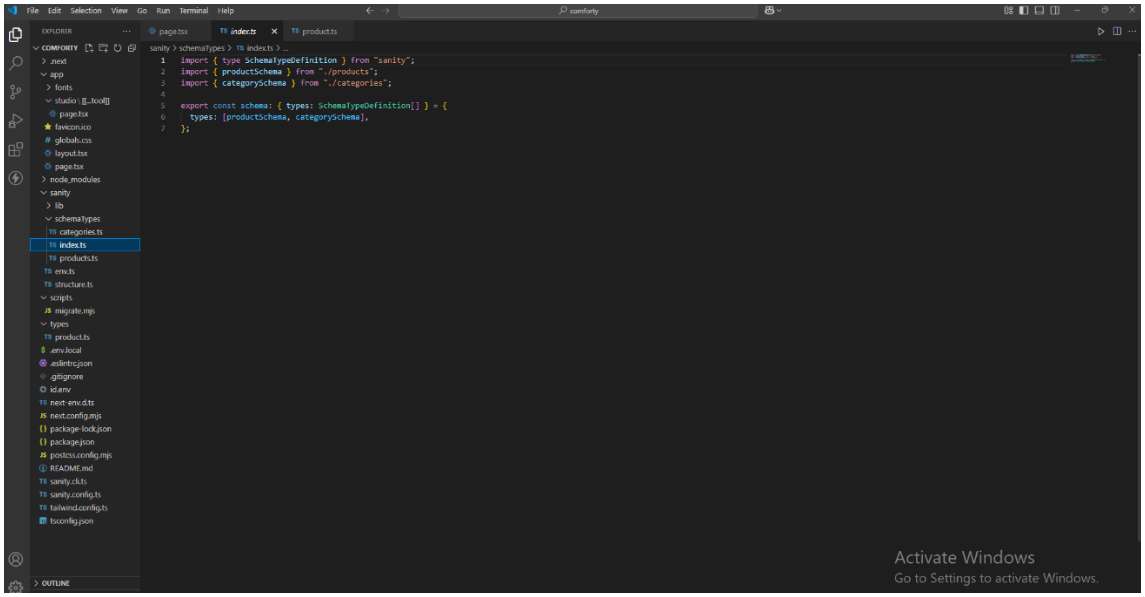
```
import { defineType } from "sanity";

export const categorySchema = defineType({
    name: 'categories',
    title: 'Categories',
    type: 'document',
    fields: [
        {
            name: 'title',
            title: 'Category Title',
            type: 'string',
        },
        {
            name: 'image',
            title: 'Category Image',
            type: 'image',
        },
        {
            title: 'Number of Products',
            name: 'products',
            type: 'number',
        }
    ],
});
```

In result on our studio:



Step-4:

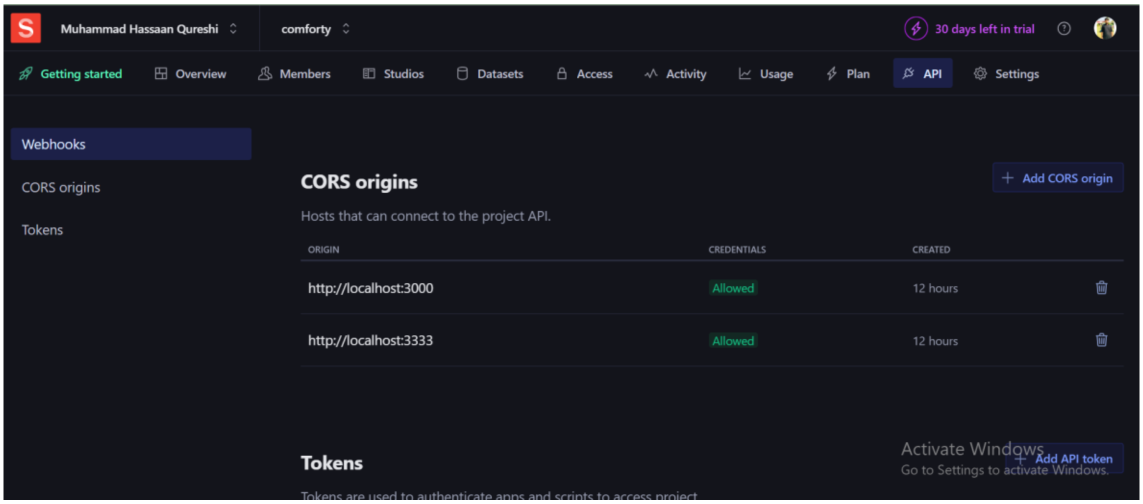
Step 4 was to import our schemas to our index.ts file.



```ts
import { type SchemaTypeDefinition } from "sanity";
import { productSchema } from "./products";
import { categorySchema } from "./categories";

export const schema: { types: SchemaTypeDefinition[] } = {
  types: [productSchema, categorySchema],
};
```

Step-5:

In our sanity project from sanity website we added our project path ex: (http://localhost3000) in CORS origins under API section.



CORS origins

Hosts that can connect to the project API.

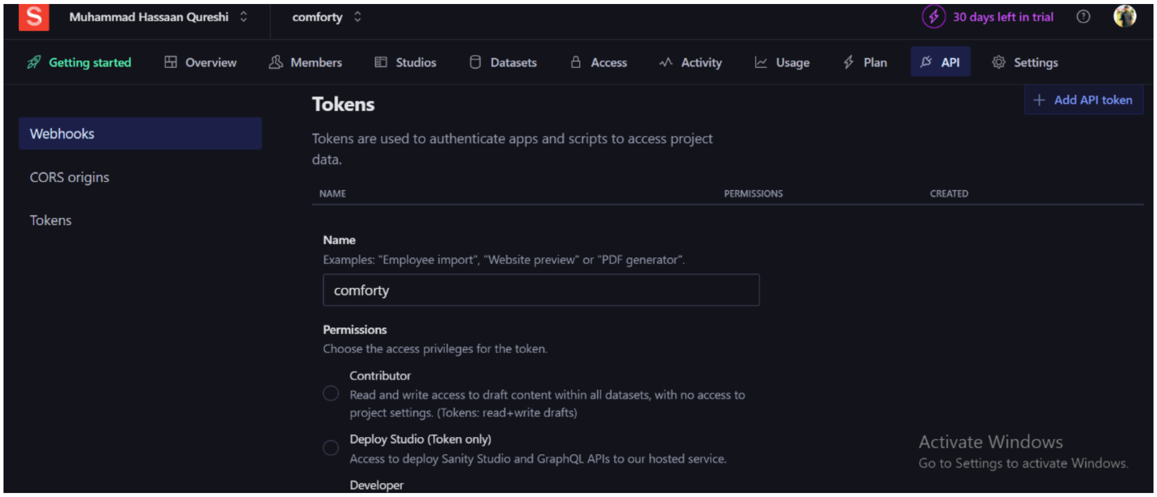
ORIGIN	CREDENTIALS	CREATED
http://localhost:3000	Allowed	12 hours
http://localhost:3333	Allowed	12 hours

Tokens

Tokens are used to authenticate apps and scripts to access project

Step-6:

Below CORS origins we generate our aut token.



Tokens

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

NAME	PERMISSIONS	CREATED

Name

Examples: "Employee import", "Website preview" or "PDF generator".

comforty

Permissions

Choose the access privileges for the token.

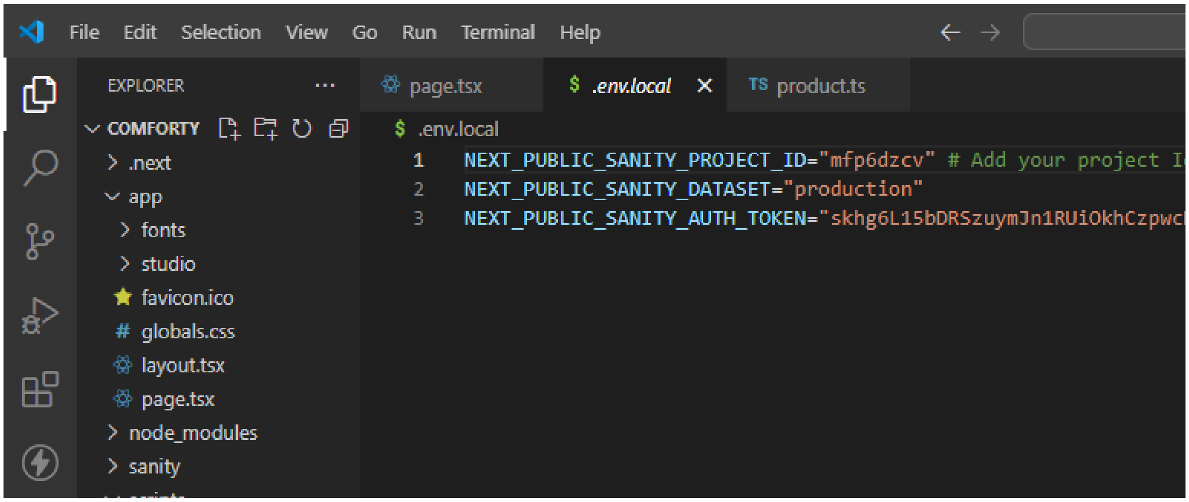
○ Contributor
Read and write access to draft content within all datasets, with no access to project settings. (Tokens: read+write drafts)

○ Deploy Studio (Token only)
Access to deploy Sanity Studio and GraphQL APIs to our hosted service.

Developer

Step-7:

We add our project id and aut token that we generated above in our .env.local file or create another .env file and include id and aut token there.

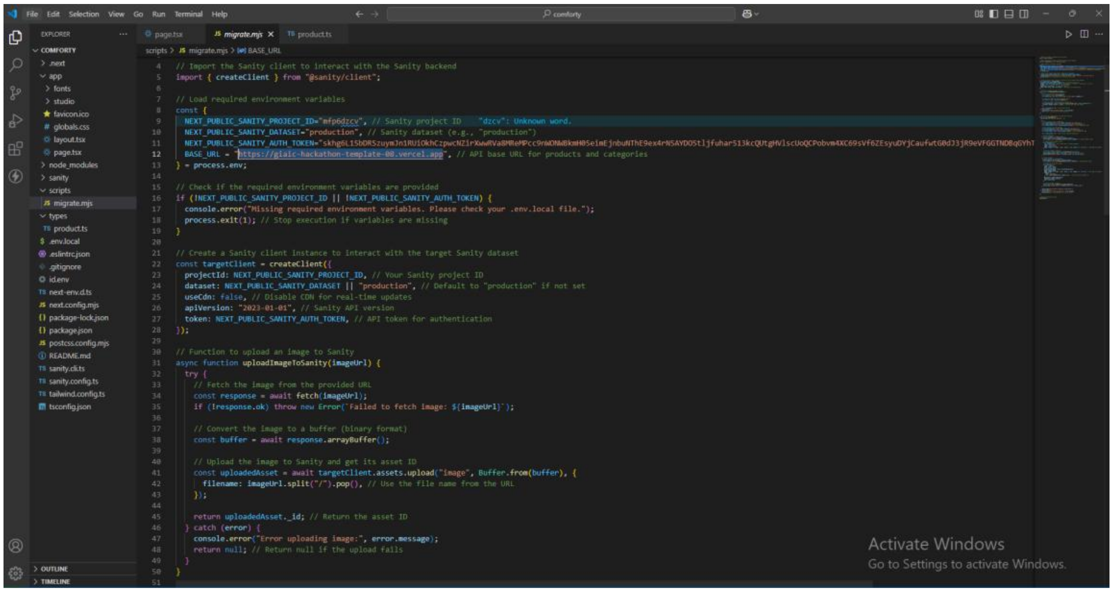


```
NEXT_PUBLIC_SANITY_PROJECT_ID="mfp6dzcv" # Add your project I
NEXT_PUBLIC_SANITY_DATASET="production"
NEXT_PUBLIC_SANITY_AUTH_TOKEN="skhg6L15bDRSzuymJn1RUiOkhCzpwc
```

Step-8:

Then we created a script folder on our root that include our migrate script that migrate our data from API to sanity.

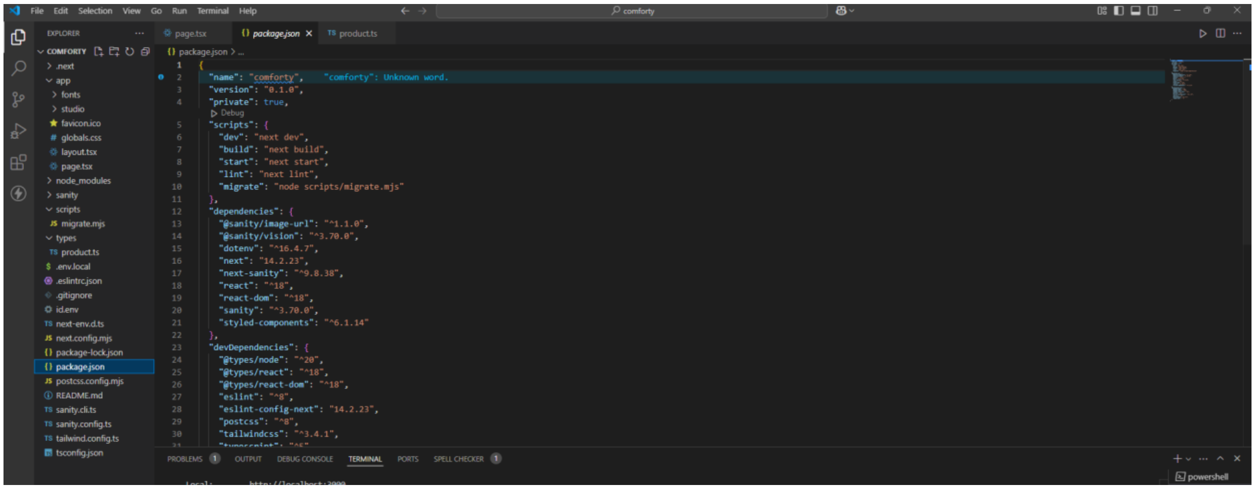
Code taken from sir Hamzah syed File.



```
// Import the Sanity client to interact with the Sanity backend
import { createClient } from "@sanity/client";

// Load required environment variables
const {
  NEXT_PUBLIC_SANITY_PROJECT_ID="mfp6dzcv", // Sanity project ID    "dzcv": Unknown word.
  NEXT_PUBLIC_SANITY_DATASET="production", // Sanity dataset (e.g., "production")
  NEXT_PUBLIC_SANITY_AUTH_TOKEN="skhg6L15bDR5zuymJn1RU1OkhCzpwcNZirXwwRVa8MReMPcc9nWONWBkmH05eimEjnbuNThE9ex4rN5AYDO5tljfuhar513kcQUtgHVlscUoQCPobvm4XC69sVf6ZEsyuDYjCaufwtG0dJ3jR9eVFGGTNDBqGYhT
  BASE_URL = "https://giaic-hackathon-template-08.vercel.app", // API base URL for products and categories
} = process.env;

// Check if the required environment variables are provided
if (!NEXT_PUBLIC_SANITY_PROJECT_ID || !NEXT_PUBLIC_SANITY_AUTH_TOKEN) {
  console.error("Missing required environment variables. Please check your .env.local file.");
  process.exit(1); // Stop execution if variables are missing
}

// Create a Sanity client instance to interact with the target Sanity dataset
const targetClient = createClient({
  projectId: NEXT_PUBLIC_SANITY_PROJECT_ID, // Your Sanity project ID
  dataset: NEXT_PUBLIC_SANITY_DATASET || "production", // Default to "production" if not set
  useCdn: false, // Disable CDN for real-time updates
  apiVersion: "2023-01-01", // Sanity API version
  token: NEXT_PUBLIC_SANITY_AUTH_TOKEN, // API token for authentication
});

// Function to upload an image to Sanity
async function uploadImageToSanity(imageUrl) {
  try {
    // Fetch the image from the provided URL
    const response = await fetch(imageUrl);
    if (!response.ok) throw new Error(`Failed to fetch image: ${imageUrl}`);

    // Convert the image to a buffer (binary format)
    const buffer = await response.arrayBuffer();

    // Upload the image to Sanity and get its asset ID
    const uploadedAsset = await targetClient.assets.upload("image", Buffer.from(buffer), {
      filename: imageUrl.split("/").pop(), // Use the file name from the URL
    });

    return uploadedAsset._id; // Return the asset ID
  } catch (error) {
    console.error("Error uploading image:", error.message);
    return null; // Return null if the upload fails
  }
}
```

Step-9:

In pakage.json file in scripts section include this line: “migrate”:”node scripts/migrate.mjs”



```
{
  "name": "comforty",    "comforty": Unknown word.
  "version": "0.1.0",
  "private": true,
  ▷ Debug
  "scripts": {
    "dev": "next dev",
    "build": "next build",
    "start": "next start",
    "lint": "next lint",
    "migrate": "node scripts/migrate.mjs"
  },
  "dependencies": {
    "@sanity/image-url": "^1.1.0",
    "@sanity/vision": "^3.70.0",
    "dotenv": "^16.4.7",
    "next": "14.2.23",
    "next-sanity": "^9.8.38",
    "react": "^18",
    "react-dom": "^18",
    "sanity": "^3.70.0",
    "styled-components": "^6.1.14"
  },
  "devDependencies": {
    "@types/node": "^20",
    "@types/react": "^18",
    "@types/react-dom": "^18",
    "eslint": "^8",
    "eslint-config-next": "14.2.23",
    "postcss": "^8",
    "tailwindcss": "^3.4.1",
```

Prepared by: Muhammad Hassaan Qureshi

Step-10:

Before running the migrate script run this command:

“npm install dotenv”

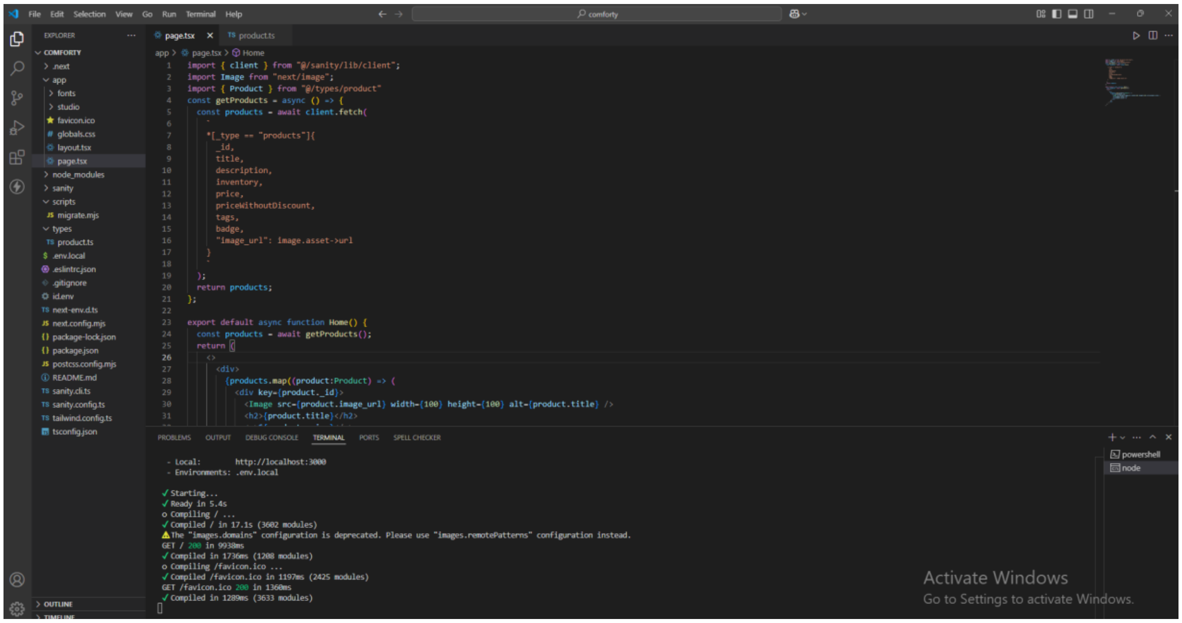
and then run the migrate script using:

“npm run migrate”

This will migrate your data from API to sanity.

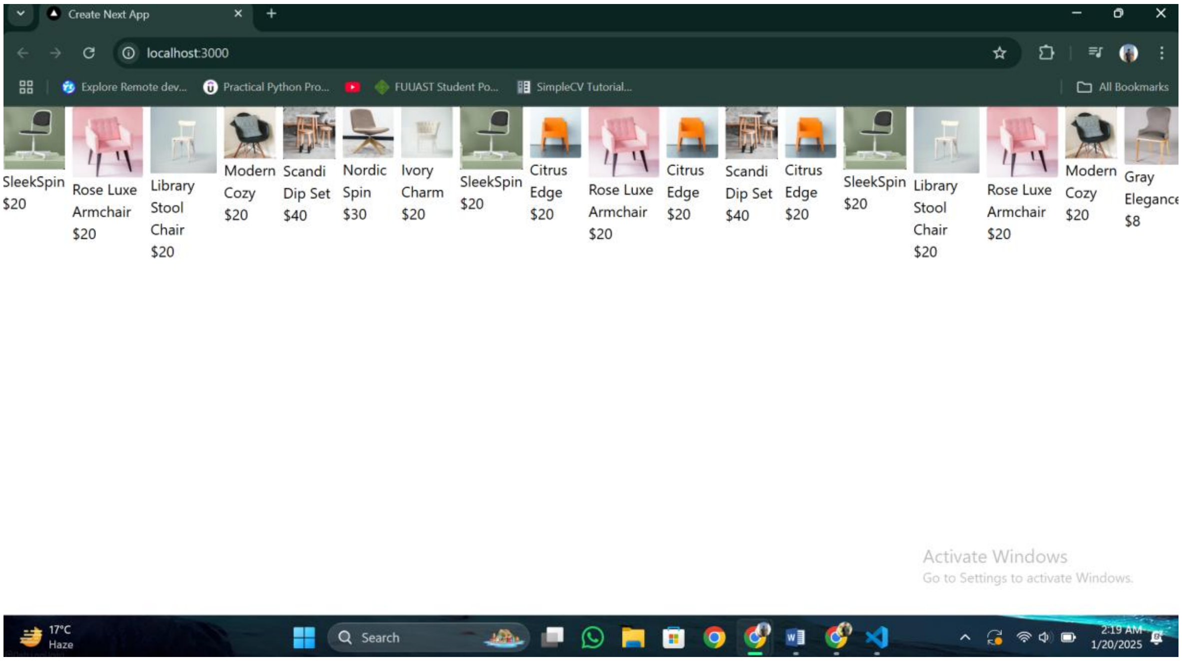
Step-11:

Now using sanity queries fetch data and use it as you like.



```
import { client } from "@/sanity/lib/client";
import Image from "next/image";
import { Product } from "@/types/product"
const getProducts = async () => {
  const products = await client.fetch(
    `
    *[_type == "products"]{
      _id,
      title,
      description,
      inventory,
      price,
      priceWithoutDiscount,
      tags,
      badge,
      "image_url": image.asset->url
    }
    `
  );
  return products;
};

export default async function Home() {
  const products = await getProducts();
  return (
    <>
      <div>
        {products.map((product:Product) => (
          <div key={product._id}>
            <Image src={product.image_url} width={100} height={100} alt={product.title} />
            <h2>{product.title}</h2>
```

```
  - Local:        http://localhost:3000
  - Environments: .env.local

 ✓ Starting...
 ✓ Ready in 5.4s
 ○ Compiling / ...
 ✓ Compiled / in 17.1s (3602 modules)
 ⚠ The "images.domains" configuration is deprecated. Please use "images.remotePatterns" configuration instead.
 GET / 200 in 9938ms
 ✓ Compiled in 1736ms (1208 modules)
 ○ Compiling /favicon.ico ...
 ✓ Compiled /favicon.ico in 1197ms (2425 modules)
 GET /favicon.ico 200 in 1360ms
 ✓ Compiled in 1289ms (3633 modules)
```



SleekSpin	$20
Rose Luxe Armchair	$20
Library Stool Chair	$20
Modern Cozy	$20
Scandi Dip Set	$40
Nordic Spin	$30
Ivory Charm	$20
SleekSpin	$20
Citrus Edge	$20
Rose Luxe Armchair	$20
Citrus Edge	$20
Scandi Dip Set	$40
Citrus Edge	$20
SleekSpin	$20
Library Stool Chair	$20
Rose Luxe Armchair	$20
Modern Cozy	$20
Gray Elegance	$8