Prepared by

KASHAF AKRAM

HACKATHONE

Day 03

API INTEGRATION AND DATA MIGRATION REPORT

1- Api integration process

we will install sanity in our Next.js project.
Then, we will create a new project in Sanity, and use it's ID and token in our Next.js application

In the sanity folder, inside the sanity/schemaTypes folder, we will create a file named Products.ts where we will copy the schema types provided by **Sir Bilal Muhammad Khan.** Similarly**,** we will create another file named categories.ts to store the data for categories.

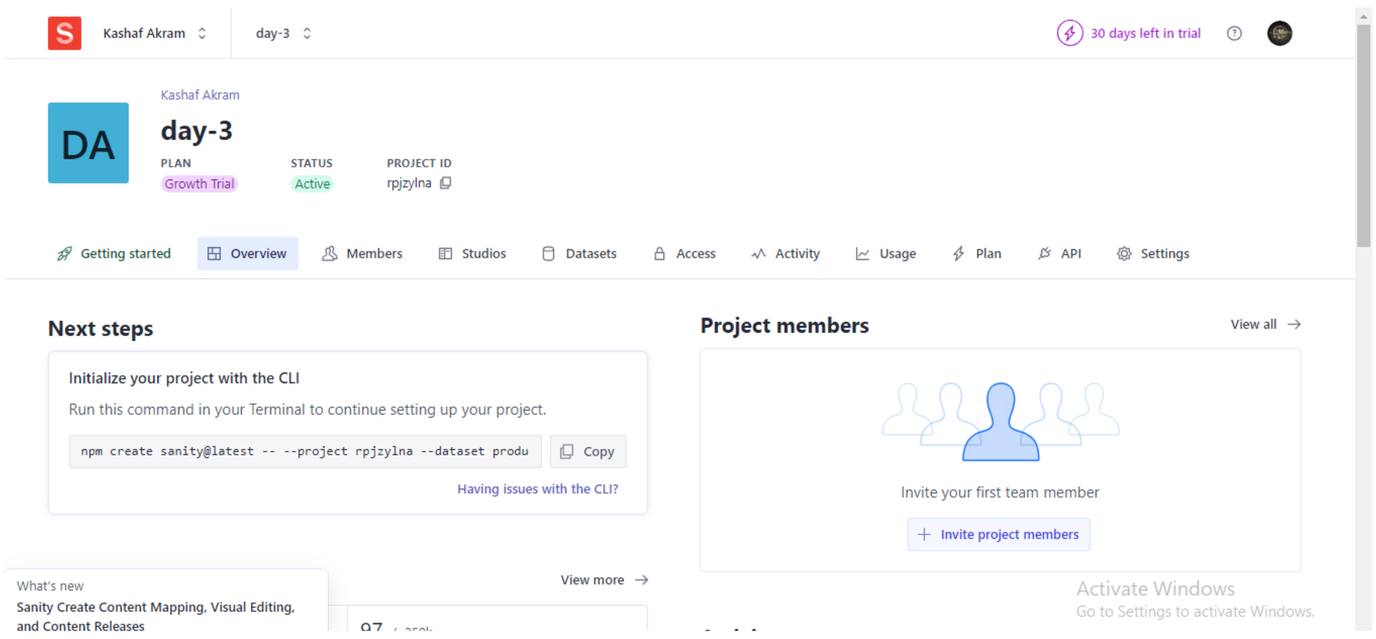
At the root, we created a folder name scripts and inside it, we created a file named importSanityData.js, where i stored the provided data

Run this command in terminal.

npm install @sanity/client axios dotenv

"import-data" : "node scripts/importSanityData.js" add this command in package.json file

run **npm run import-data**



2. Adjustment made of schemas:

The Product schema defines the structure and organization of the product data in the Sanity content platform. It is designed to store comprehensive details about each product, ensuring it is versatile and fits various use cases such as e-commerce platforms or product catalogs

3: Migration Steps and tool used.

```
import { defineQuery } from "next-sanity";

export const allProducts = defineQuery(`*[_type == "product"]{
    _id,
    name,
    description,
    "categoryName": category->title,
    "imageUrl": image.asset->url,
    price,
    quantity,
    dimensions,
    features,
    tags
    }`)
```

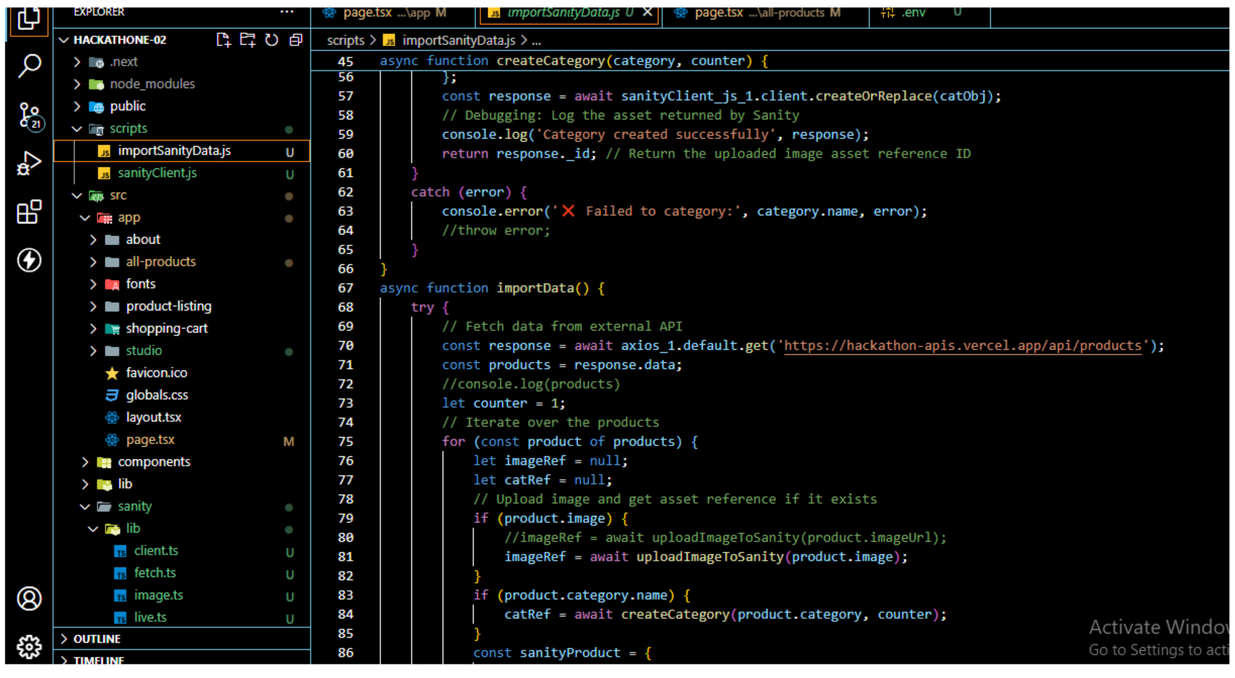
The query is designed to fetch complete and well -structured product data from theSanity backend.

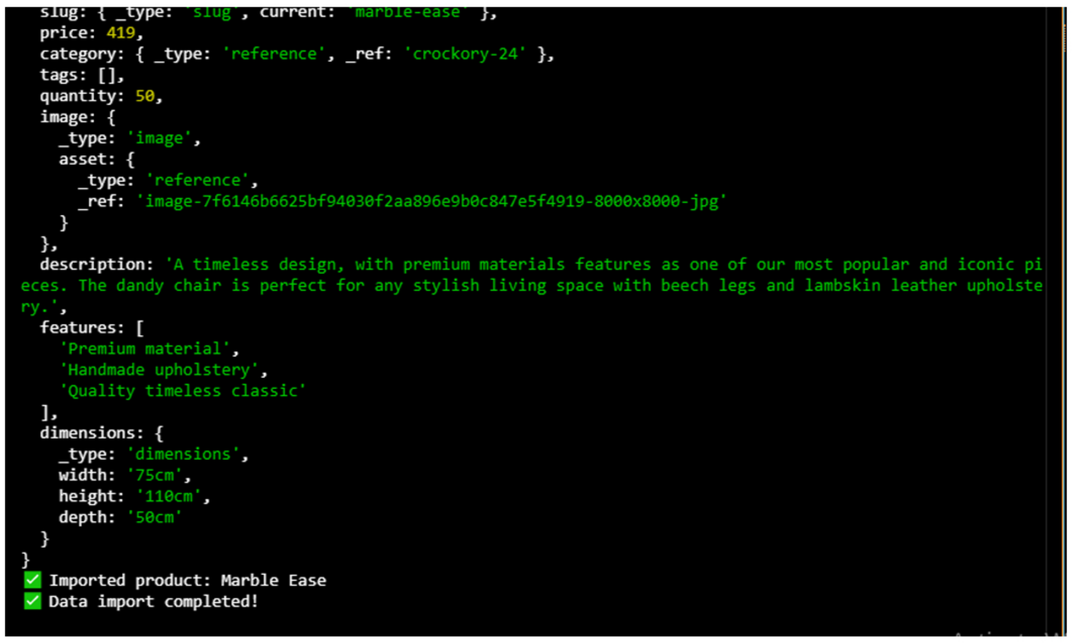
```
import { createClient } from "next-sanity"

const client = createClient({
    projectId:"rpjzylna",
    dataset: "production",
    useCdn: true,
    apiVersion: "2023-10-10"
})
export async function sanityFetch({query, params = {}}: {query:string, params?:any}){
    return await client.fetch(query, params)
}
```

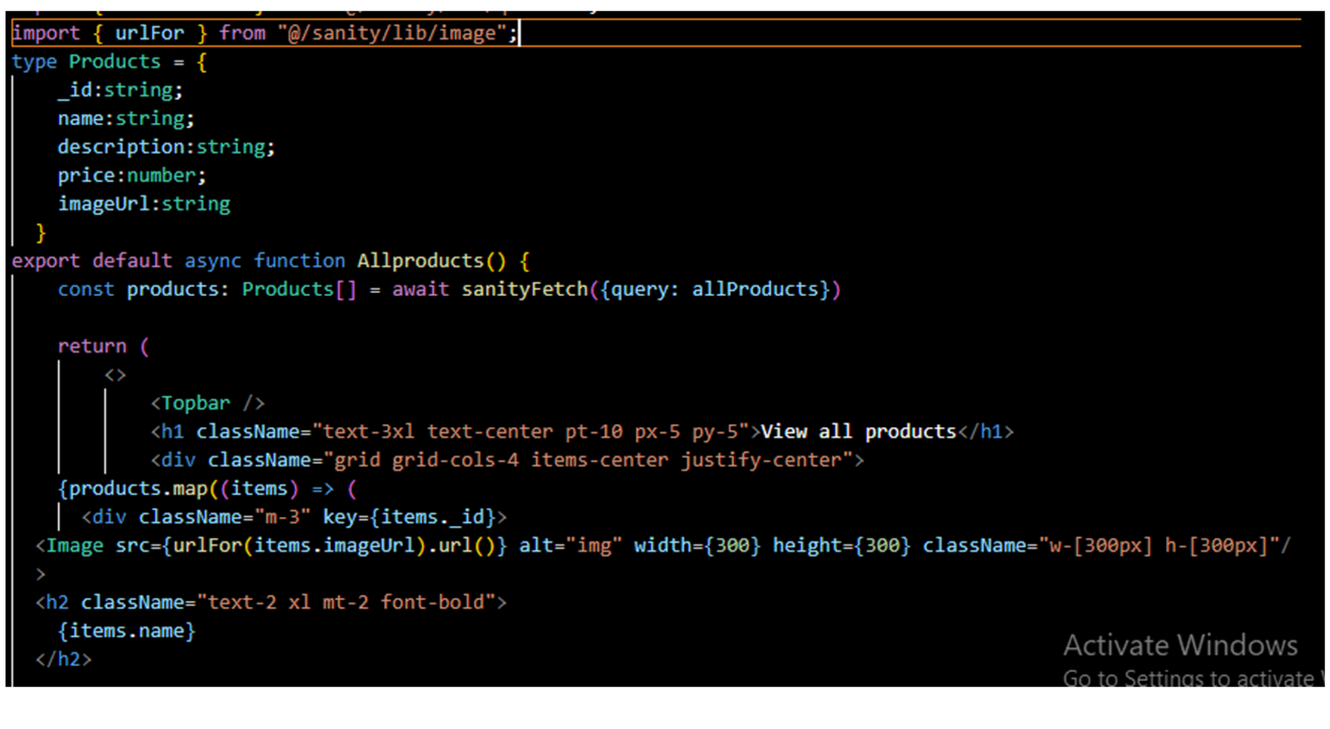
This code initializes a Sanity client using the `createClient` function from the `next-sanity` package to interact with the Sanity CMS. It connects to the project with the specified `projectId`, `dataset`, and `apiVersion`. The `sanityFetch` function is a reusable utility for querying data from Sanity by accepting a `query` string and optional `params`. It simplifies data fetching by returning the result of the `fetch` method, making it easy to retrieve and manage content dynamically in the application.

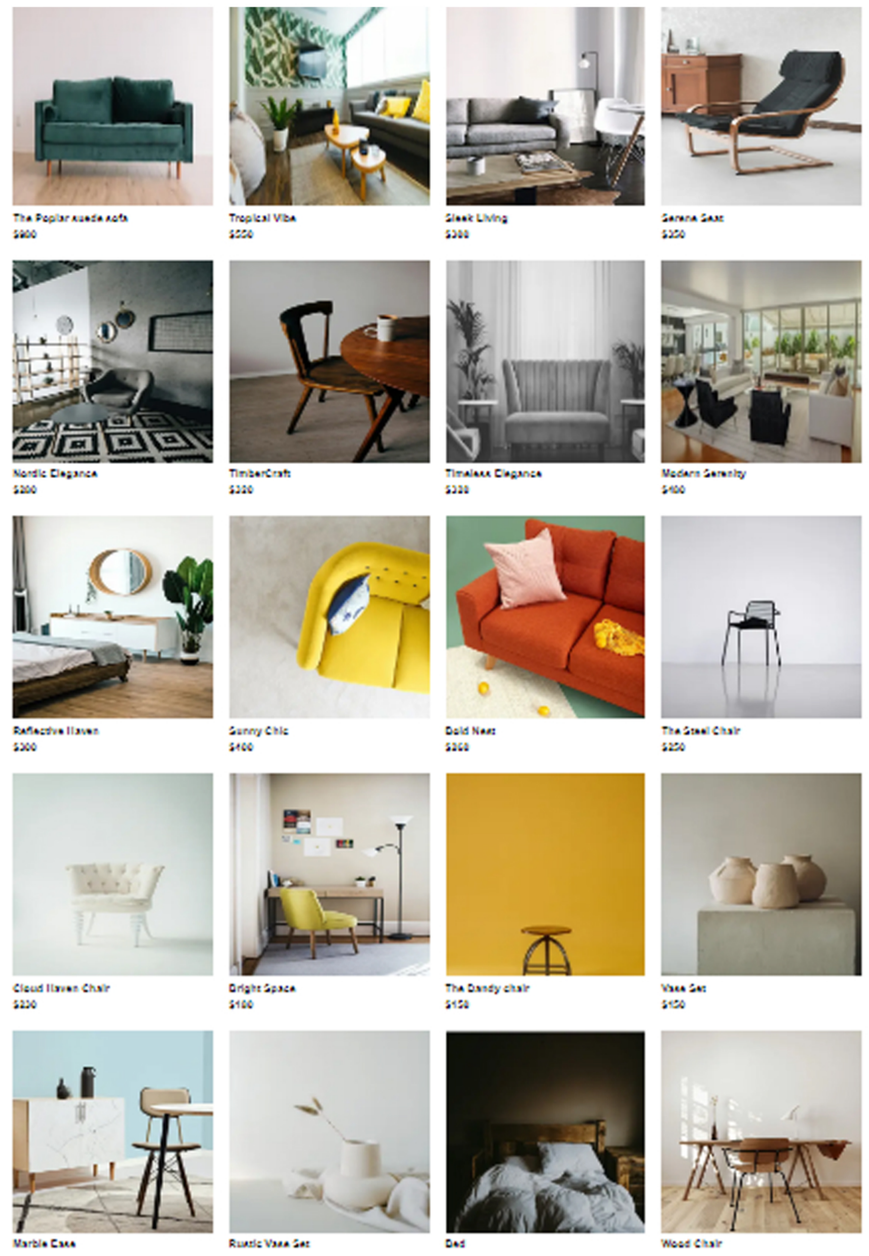
IMPORTING DATA INTO SANITY





DATA SUCCESSFULLY DISPLAYED IN THE FRONTEND





This code is a frontend component for displaying all products retrieved from Sanity CMS. It uses the sanityFetch utility to fetch product data with the allProducts query. The data is displayed in a grid layout, with each product showing its image, name, and price. The component also includes a hero section with promotional content and a sign-up form for newsletters. The Topbar and Footer components are used for consistent navigation and page structure, providing a user-friendly interface for viewing and interacting with the products.

Day 3 Checklist:

Self-Validation Checklist:

API Understanding: ✔

API Integration in Next.js: ✔

Schema Validation: ✔

Submission Preparation: ✔

Data Migration: ✔