DAY_03 API INTEGRATION AND DATA MIGRATION

MARKETPLACE_BUILDER_HACKATHON_AVION_(E-COMMERCE)

1. INTRODUCTION:

This comprehensive report includes all step by step methods and procedure of API integration and Data Migration from **API** to **Sanity CMS.** Which will provide AVION a backend for the data as well as the information storage.

Key Points:

- API Integration
- Data Migration
- Displaying data dynamically on frontend.

2. API Integration

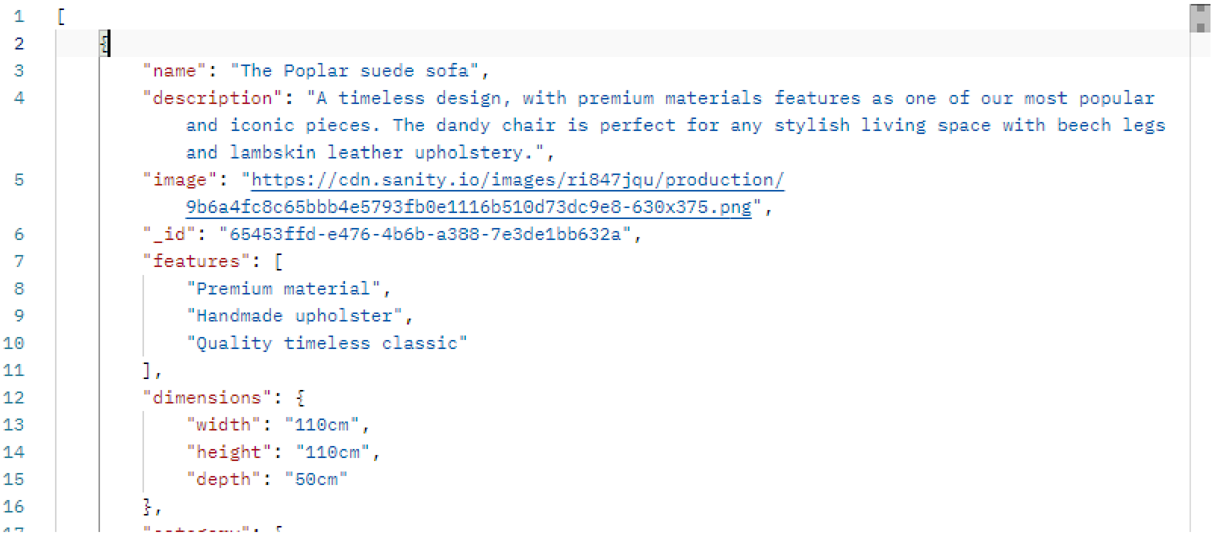
Fetching data from API:

Provide API for Template 2: https://hackathon-apis.vercel.app/api/products

EndPoints: products

Integrated Fields:

- _id: Unique id for every product.
- Name: Name of every product.
- Price: Price of every product.
- Description: Product descriptions includes essential information.
- Dimensions: Products Dimensions.
- Image: Reference Image.
- Tags: Special Key words.
- Category: The category of products (Chairs, table, wooden etc.).

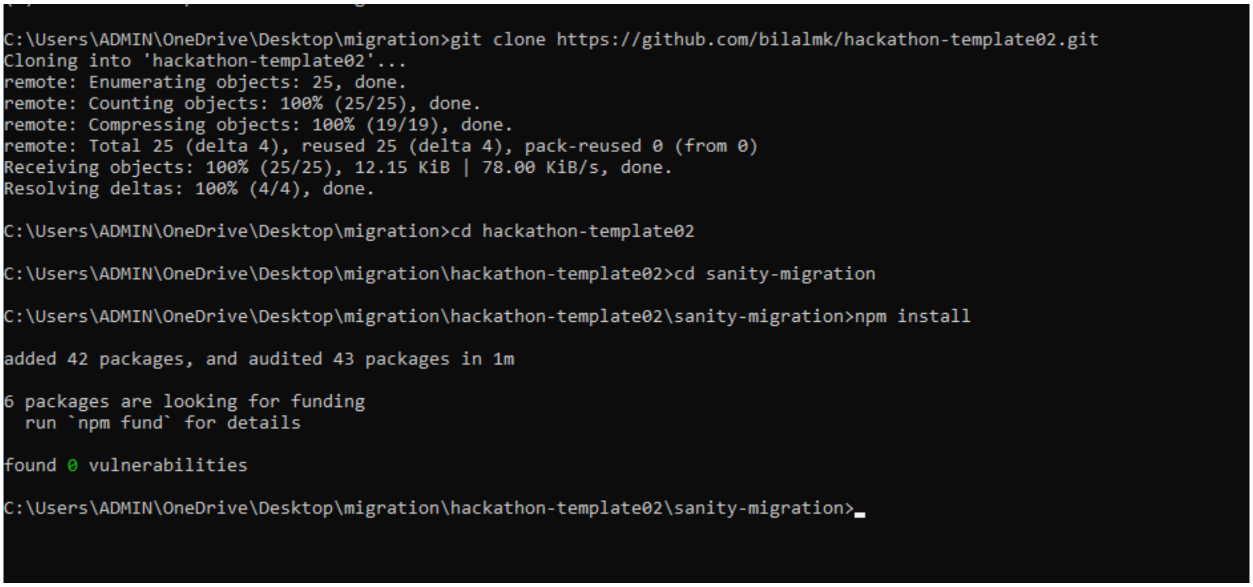


```
[
    {
        "name": "The Poplar suede sofa",
        "description": "A timeless design, with premium materials features as one of our most popular
            and iconic pieces. The dandy chair is perfect for any stylish living space with beech legs
            and lambskin leather upholstery.",
        "image": "https://cdn.sanity.io/images/ri847jqu/production/
            9b6a4fc8c65bbb4e5793fb0e1116b510d73dc9e8-630x375.png",
        "_id": "65453ffd-e476-4b6b-a388-7e3de1bb632a",
        "features": [
            "Premium material",
            "Handmade upholster",
            "Quality timeless classic"
        ],
        "dimensions": {
            "width": "110cm",
            "height": "110cm",
            "depth": "50cm"
        },
        "category": [
```

DATA MIGRATION:

Data migration Scripts is provided according to the Template by Honorable Faculty.

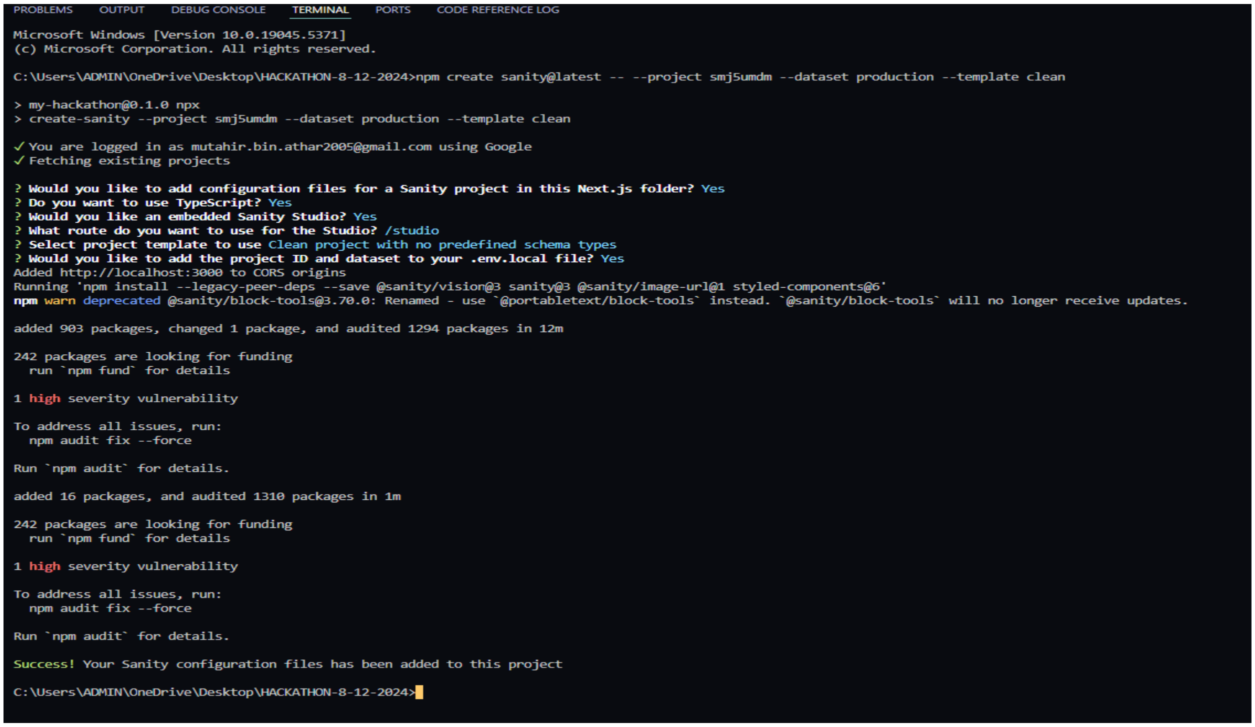
https://github.com/bilalmk/hackathon-template02/tree/main/sanity-migration



```
C:\Users\ADMIN\OneDrive\Desktop\migration>git clone https://github.com/bilalmk/hackathon-template02.git
Cloning into 'hackathon-template02'...
remote: Enumerating objects: 25, done.
remote: Counting objects: 100% (25/25), done.
remote: Compressing objects: 100% (19/19), done.
remote: Total 25 (delta 4), reused 25 (delta 4), pack-reused 0 (from 0)
Receiving objects: 100% (25/25), 12.15 KiB | 78.00 KiB/s, done.
Resolving deltas: 100% (4/4), done.

C:\Users\ADMIN\OneDrive\Desktop\migration>cd hackathon-template02

C:\Users\ADMIN\OneDrive\Desktop\migration\hackathon-template02>cd sanity-migration

C:\Users\ADMIN\OneDrive\Desktop\migration\hackathon-template02\sanity-migration>npm install

added 42 packages, and audited 43 packages in 1m

6 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities

C:\Users\ADMIN\OneDrive\Desktop\migration\hackathon-template02\sanity-migration>
```

Creating Sanity Project:

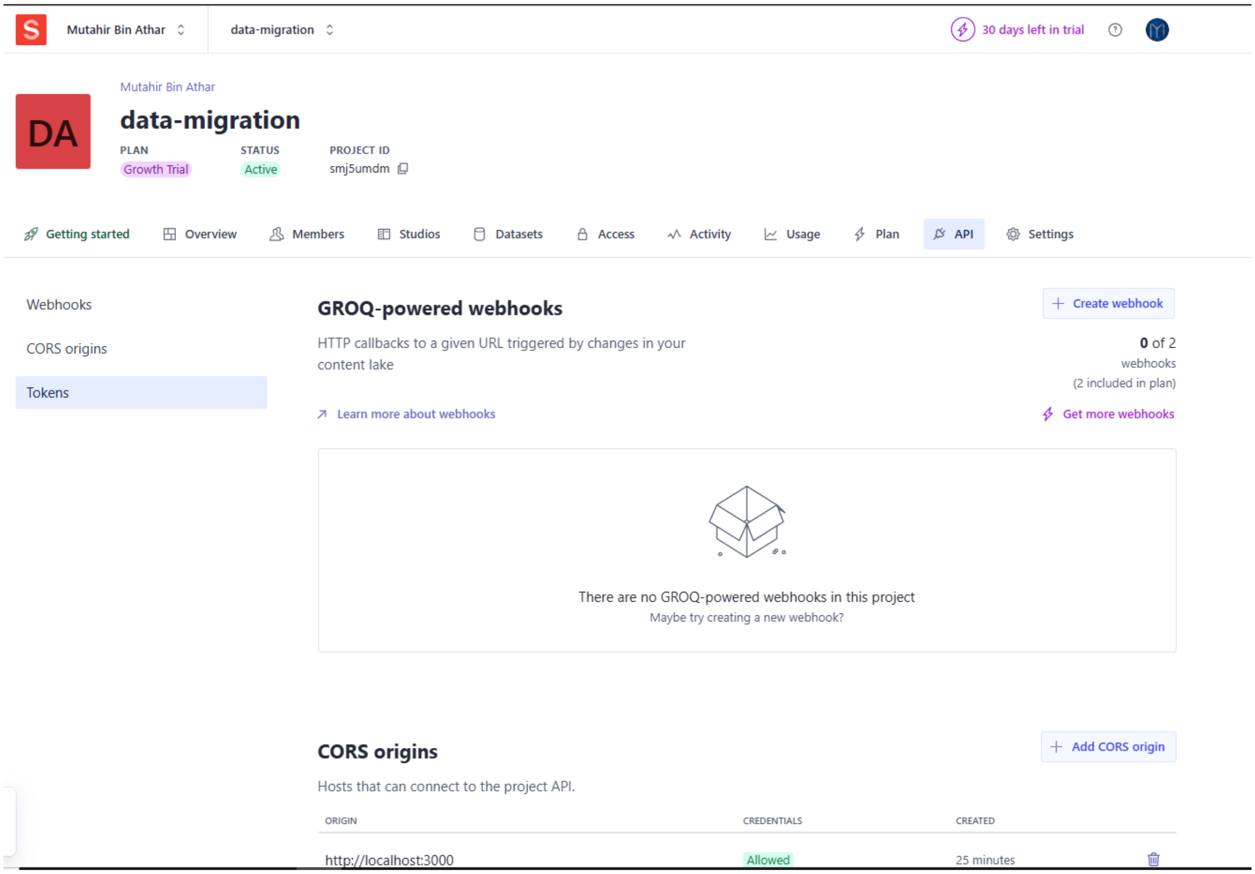
Creating Sanity Project using CLI command.



```
Microsoft Windows [Version 10.0.19045.5371]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ADMIN\OneDrive\Desktop\HACKATHON-8-12-2024>npm create sanity@latest -- --project smj5umdm --dataset production --template clean

> my-hackathon@0.1.0 npx
> create-sanity --project smj5umdm --dataset production --template clean

✓ You are logged in as mutahir.bin.athar2005@gmail.com using Google
✓ Fetching existing projects

? Would you like to add configuration files for a Sanity project in this Next.js folder? Yes
? Do you want to use TypeScript? Yes
? Would you like an embedded Sanity Studio? Yes
? What route do you want to use for the Studio? /studio
? Select project template to use Clean project with no predefined schema types
? Would you like to add the project ID and dataset to your .env.local file? Yes
Added http://localhost:3000 to CORS origins
Running 'npm install --legacy-peer-deps --save @sanity/vision@3 sanity@3 @sanity/image-url@1 styled-components@6'
npm warn deprecated @sanity/block-tools@3.70.0: Renamed - use `@portabletext/block-tools` instead. `@sanity/block-tools` will no longer receive updates.

added 903 packages, changed 1 package, and audited 1294 packages in 12m

242 packages are looking for funding
  run `npm fund` for details

1 high severity vulnerability

To address all issues, run:
  npm audit fix --force

Run `npm audit` for details.

added 16 packages, and audited 1310 packages in 1m

242 packages are looking for funding
  run `npm fund` for details

1 high severity vulnerability

To address all issues, run:
  npm audit fix --force

Run `npm audit` for details.

Success! Your Sanity configuration files has been added to this project

C:\Users\ADMIN\OneDrive\Desktop\HACKATHON-8-12-2024>
```

Sanity Project Created:

Creating Sanity Project using CLI command.



Mutahir Bin Athar

data-migration

PLAN: Growth Trial
STATUS: Active
PROJECT ID: smj5umdm

GROQ-powered webhooks

HTTP callbacks to a given URL triggered by changes in your content lake

0 of 2 webhooks (2 included in plan)

There are no GROQ-powered webhooks in this project
Maybe try creating a new webhook?

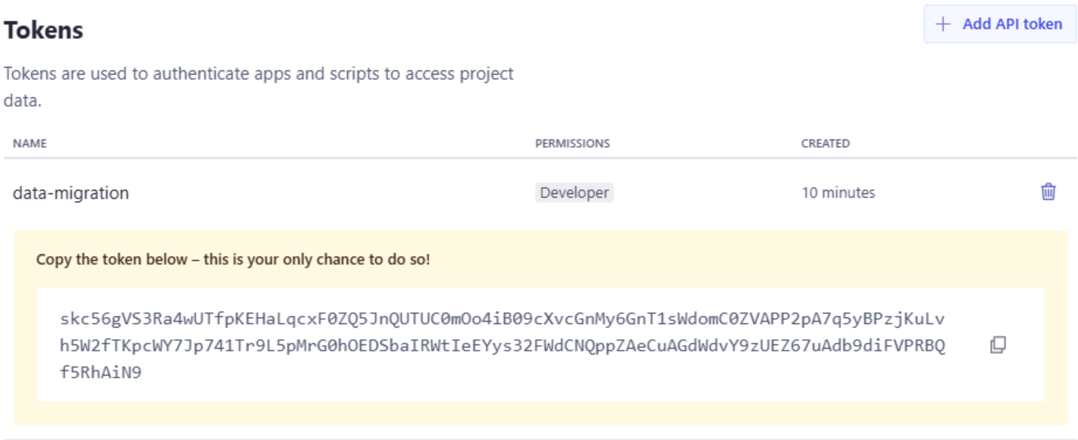
CORS origins

Hosts that can connect to the project API.

ORIGIN	CREDENTIALS	CREATED
http://localhost:3000	Allowed	25 minutes

Sanity Token:

Creating Token While Navigation from Dashboard to API and then Token an creating using developer mode.



Tokens

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

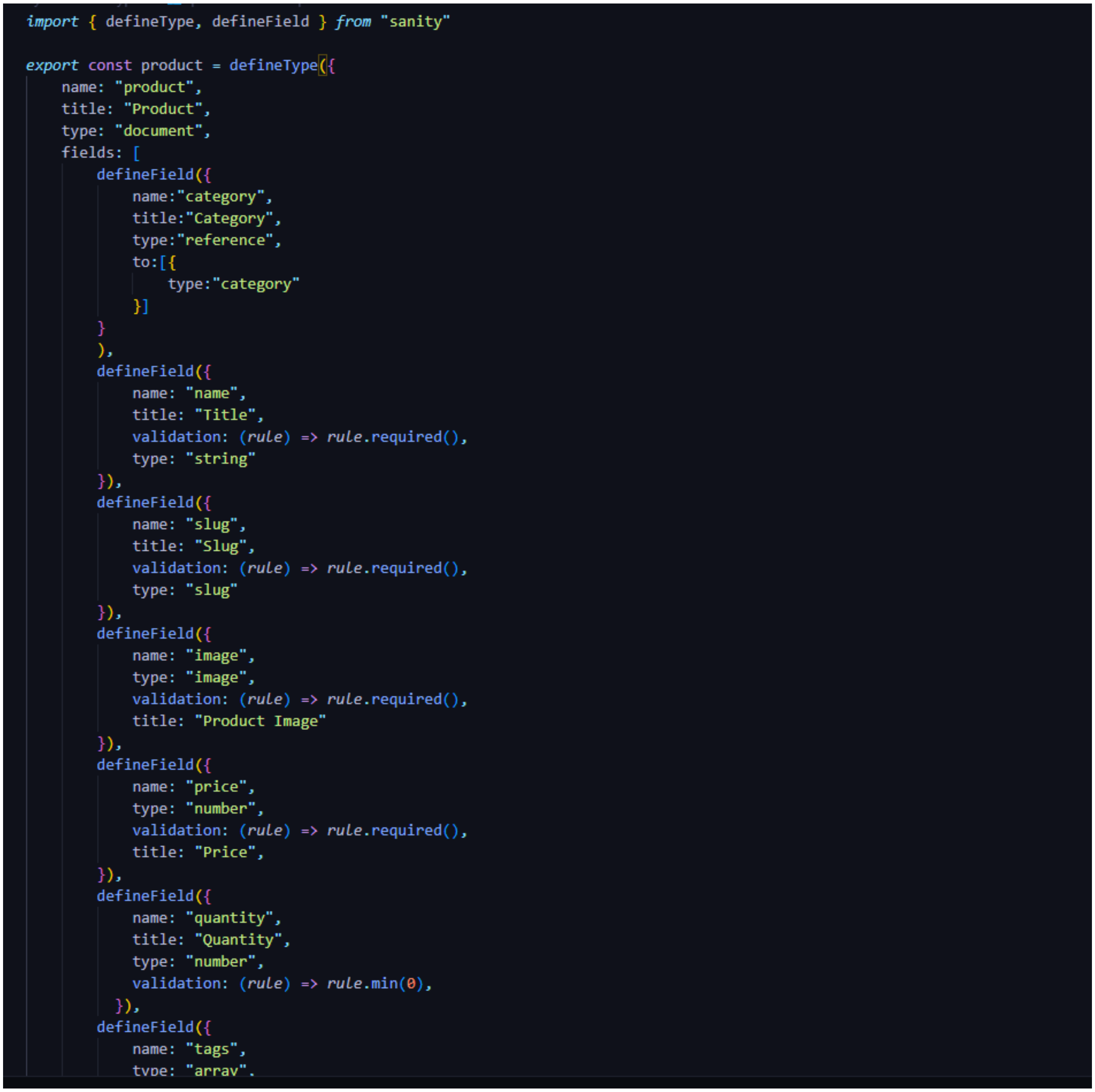
NAME	PERMISSIONS	CREATED
data-migration	Developer	10 minutes

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

```
skc56gVS3Ra4wUTfpKEHaLqcxF0ZQ5JnQUTUC0mOo4iB09cXvcGnMy6GnT1sWdomC0ZVAPP2pA7q5yBPzjKuLv
h5W2fTKpcWY7Jp741Tr9L5pMrG0hOEDSbaIRWtIeEYys32FWdCNQppZAeCuAGdWdvY9zUEZ67uAdb9diFVPRBQ
f5RhAiN9
```

Sanity Schema:

Putting Provided sanity Schema in schematypes/product.ts:

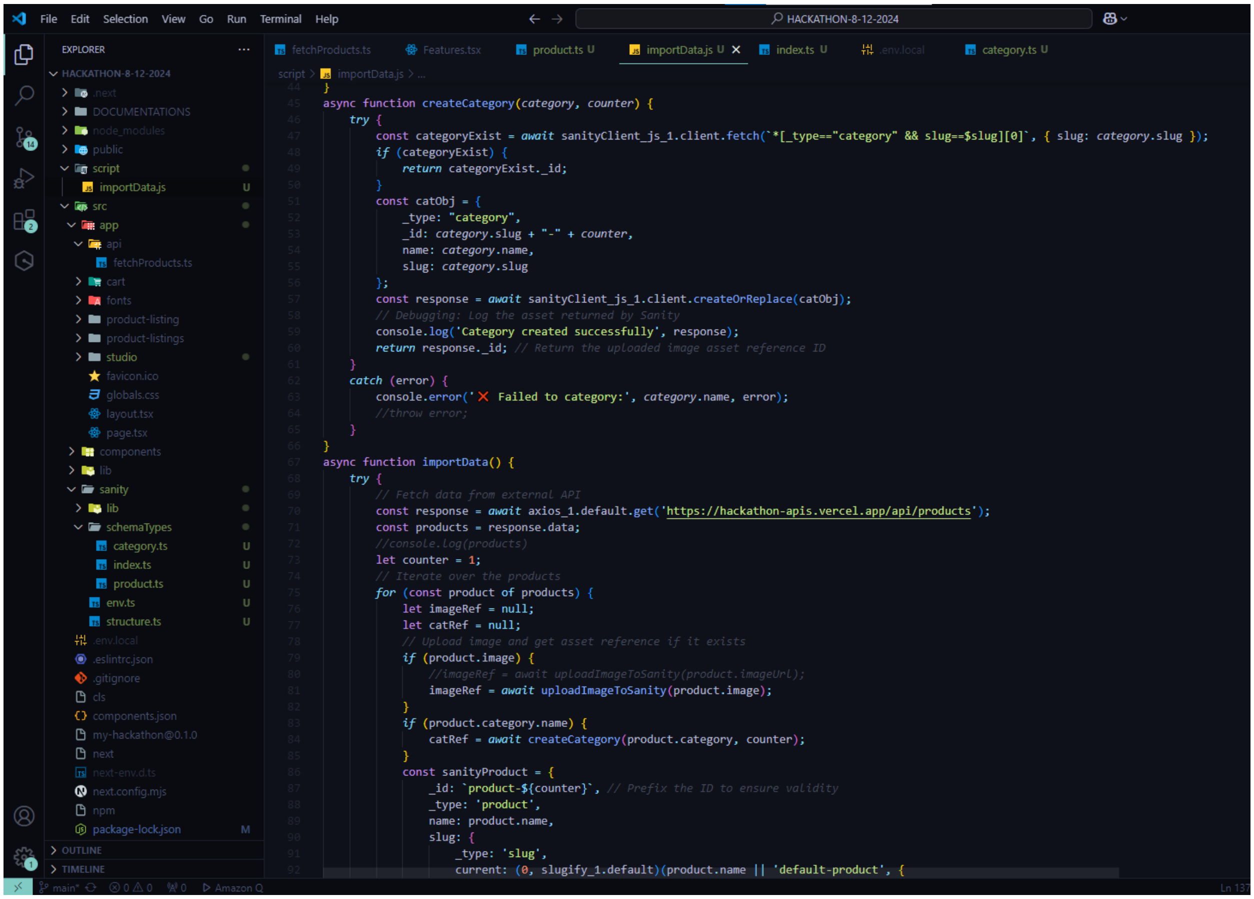


```
import { defineType, defineField } from "sanity"

export const product = defineType({
    name: "product",
    title: "Product",
    type: "document",
    fields: [
        defineField({
            name:"category",
            title:"Category",
            type:"reference",
            to:[{
                type:"category"
            }]
        }
        ),
        defineField({
            name: "name",
            title: "Title",
            validation: (rule) => rule.required(),
            type: "string"
        }),
        defineField({
            name: "slug",
            title: "Slug",
            validation: (rule) => rule.required(),
            type: "slug"
        }),
        defineField({
            name: "image",
            type: "image",
            validation: (rule) => rule.required(),
            title: "Product Image"
        }),
        defineField({
            name: "price",
            type: "number",
            validation: (rule) => rule.required(),
            title: "Price",
        }),
        defineField({
            name: "quantity",
            title: "Quantity",
            type: "number",
            validation: (rule) => rule.min(0),
          }),
        defineField({
            name: "tags",
            type: "array",
```

importData.js:

By compiling this file the data of the API will migrate to the Sanity CMS.

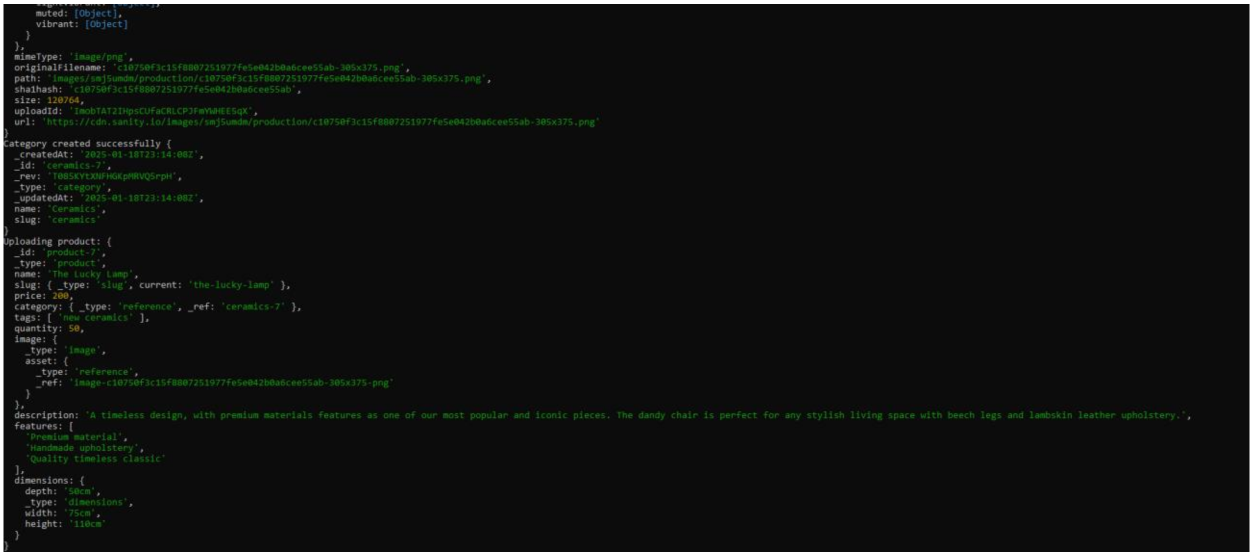
" Node importData.js "



```javascript
}
async function createCategory(category, counter) {
    try {
        const categoryExist = await sanityClient_js_1.client.fetch(`*[_type=="category" && slug==$slug][0]`, { slug: category.slug });
        if (categoryExist) {
            return categoryExist._id;
        }
        const catObj = {
            _type: "category",
            _id: category.slug + "-" + counter,
            name: category.name,
            slug: category.slug
        };
        const response = await sanityClient_js_1.client.createOrReplace(catObj);
        // Debugging: Log the asset returned by Sanity
        console.log('Category created successfully', response);
        return response._id; // Return the uploaded image asset reference ID
    }
    catch (error) {
        console.error('❌ Failed to category:', category.name, error);
        //throw error;
    }
}
async function importData() {
    try {
        // Fetch data from external API
        const response = await axios_1.default.get('https://hackathon-apis.vercel.app/api/products');
        const products = response.data;
        //console.log(products)
        let counter = 1;
        // Iterate over the products
        for (const product of products) {
            let imageRef = null;
            let catRef = null;
            // Upload image and get asset reference if it exists
            if (product.image) {
                //imageRef = await uploadImageToSanity(product.imageUrl);
                imageRef = await uploadImageToSanity(product.image);
            }
            if (product.category.name) {
                catRef = await createCategory(product.category, counter);
            }
            const sanityProduct = {
                _id: `product-${counter}`, // Prefix the ID to ensure validity
                _type: 'product',
                name: product.name,
                slug: {
                    _type: 'slug',
                    current: (0, slugify_1.default)(product.name || 'default-product', {
```

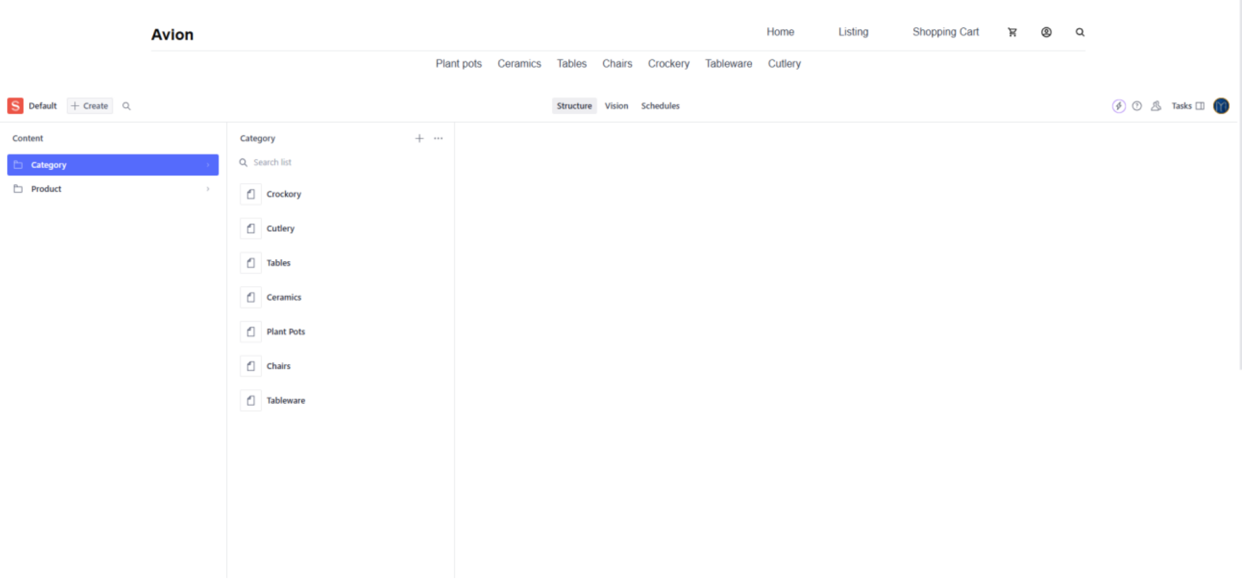
Data Migrated Successfully:

After this the data successfully started migrating to sanity.



```
      muted: [Object],
      vibrant: [Object]
    }
  },
  mimeType: 'image/png',
  originalFilename: 'c10750f3c15f8807251977fe5e042b0a6cee55ab-305x375.png',
  path: 'images/smj5umdm/production/c10750f3c15f8807251977fe5e042b0a6cee55ab-305x375.png',
  sha1hash: 'c10750f3c15f8807251977fe5e042b0a6cee55ab',
  size: 120764,
  uploadId: 'ImobTAT2IHpsCUfaCRLCPJFmYWHEESqX',
  url: 'https://cdn.sanity.io/images/smj5umdm/production/c10750f3c15f8807251977fe5e042b0a6cee55ab-305x375.png'
}
Category created successfully {
  _createdAt: '2025-01-18T23:14:08Z',
  _id: 'ceramics-7',
  _rev: 'T08SKYtXNFHGKpMRVQSrpH',
  _type: 'category',
  _updatedAt: '2025-01-18T23:14:08Z',
  name: 'Ceramics',
  slug: 'ceramics'
}
Uploading product: {
  _id: 'product-7',
  _type: 'product',
  name: 'The Lucky Lamp',
  slug: { _type: 'slug', current: 'the-lucky-lamp' },
  price: 200,
  category: { _type: 'reference', _ref: 'ceramics-7' },
  tags: [ 'new ceramics' ],
  quantity: 50,
  image: {
    _type: 'image',
    asset: {
      _type: 'reference',
      _ref: 'image-c10750f3c15f8807251977fe5e042b0a6cee55ab-305x375-png'
    }
  },
  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.',
  features: [
    'Premium material',
    'Handmade upholstery',
    'Quality timeless classic'
  ],
  dimensions: {
    depth: '50cm',
    _type: 'dimensions',
    width: '75cm',
    height: '110cm'
  }
}
```

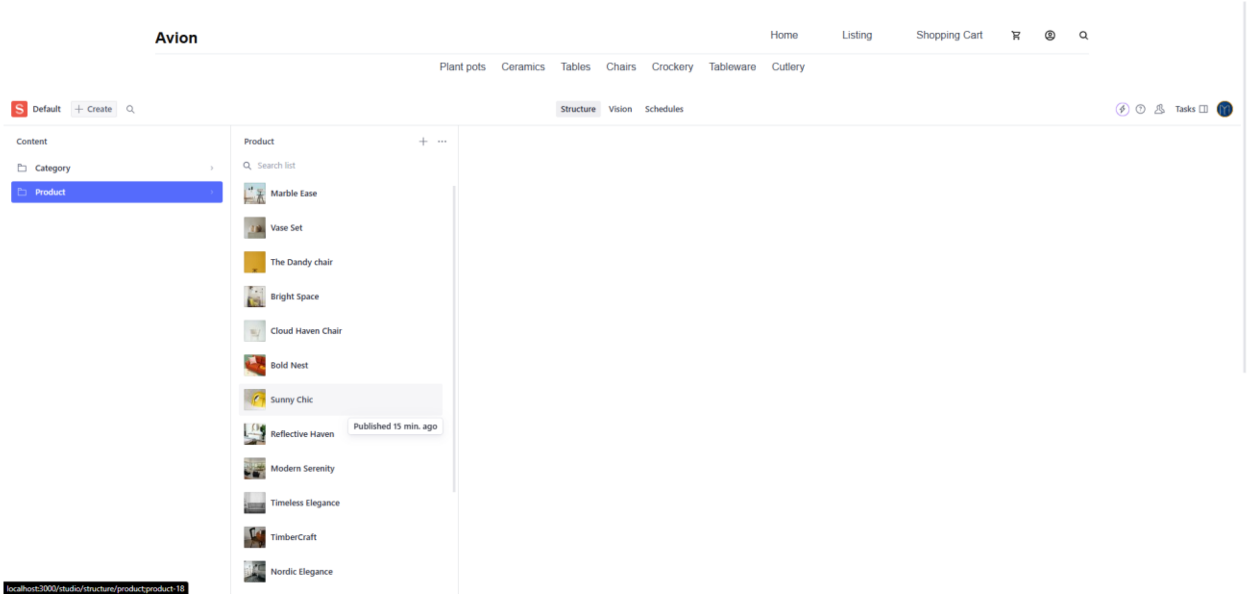
Categories Displayed on Sanity:



Category

- Crockory
- Cutlery
- Tables
- Ceramics
- Plant Pots
- Chairs
- Tableware

Products Displayed on Sanity:



Product

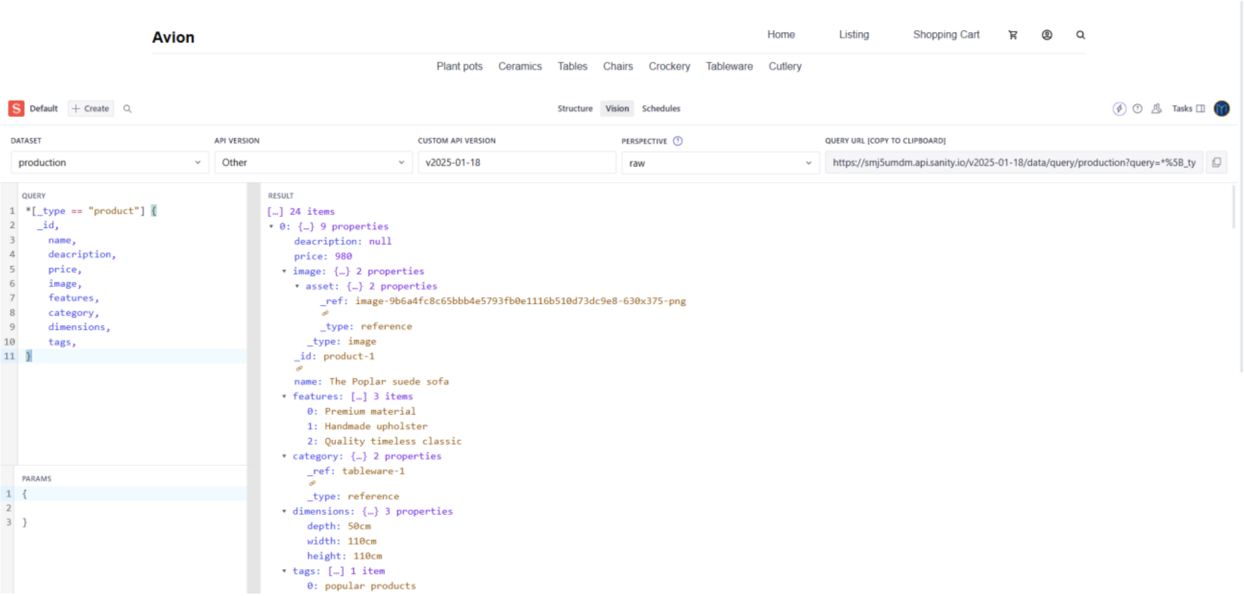
Search list

- Marble Ease
- Vase Set
- The Dandy chair
- Bright Space
- Cloud Haven Chair
- Bold Nest
- Sunny Chic
- Reflective Haven
- Modern Serenity
- Timeless Elegance
- TimberCraft
- Nordic Elegance

Published 15 min. ago

localhost:3000/studio/structure/product;product-18

Query for Fetching Data:



DATASET: production

API VERSION: Other

CUSTOM API VERSION: v2025-01-18

PERSPECTIVE: raw

QUERY URL [COPY TO CLIPBOARD]: https://smj5umdm.api.sanity.io/v2025-01-18/data/query/production?query=*%5B_ty

QUERY

```
*[_type == "product"] {
  _id,
    name,
    deacription,
    price,
    image,
    features,
    category,
    dimensions,
    tags,
}
```

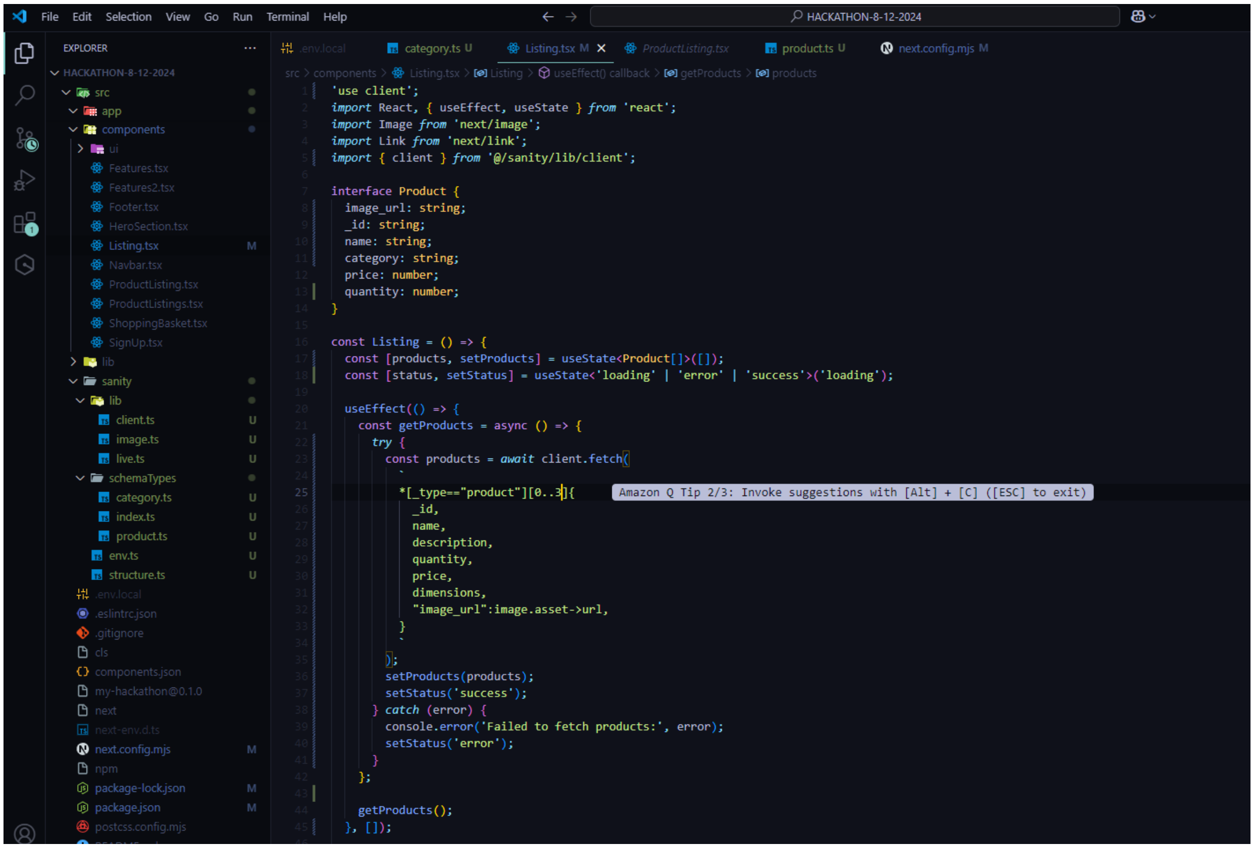
PARAMS

```
{

}
```

RESULT

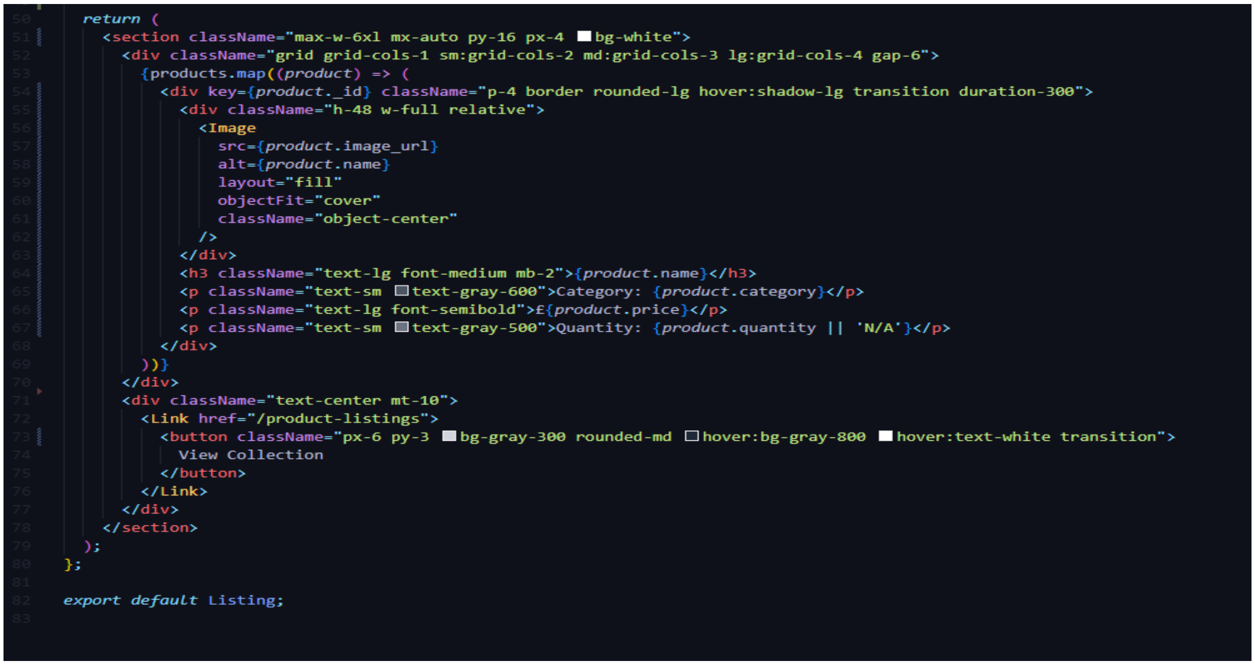
```
[…] 24 items
▾ 0: {…} 9 properties
    deacription: null
    price: 980
  ▾ image: {…} 2 properties
    ▾ asset: {…} 2 properties
        _ref: image-9b6a4fc8c65bbb4e5793fb0e1116b510d73dc9e8-630x375-png
        _type: reference
      _type: image
    _id: product-1
    name: The Poplar suede sofa
  ▾ features: […] 3 items
      0: Premium material
      1: Handmade upholster
      2: Quality timeless classic
  ▾ category: {…} 2 properties
      _ref: tableware-1
      _type: reference
  ▾ dimensions: {…} 3 properties
      depth: 50cm
      width: 110cm
      height: 110cm
  ▾ tags: […] 1 item
      0: popular products
```

Running Query in the code:



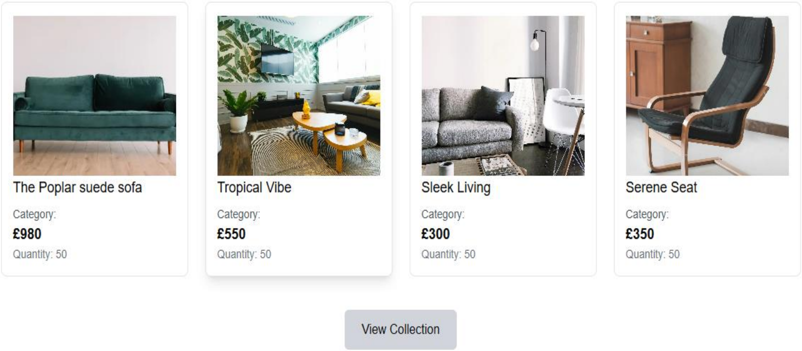
```tsx
'use client';
import React, { useEffect, useState } from 'react';
import Image from 'next/image';
import Link from 'next/link';
import { client } from '@/sanity/lib/client';

interface Product {
  image_url: string;
  _id: string;
  name: string;
  category: string;
  price: number;
  quantity: number;
}

const Listing = () => {
  const [products, setProducts] = useState<Product[]>([]);
  const [status, setStatus] = useState<'loading' | 'error' | 'success'>('loading');

  useEffect(() => {
    const getProducts = async () => {
      try {
        const products = await client.fetch(
          `
          *[_type=="product"][0..3]{
            _id,
            name,
            description,
            quantity,
            price,
            dimensions,
            "image_url":image.asset->url,
          }
          `
        );
        setProducts(products);
        setStatus('success');
      } catch (error) {
        console.error('Failed to fetch products:', error);
        setStatus('error');
      }
    };

    getProducts();
  }, []);
```

Getting product's details:



```tsx
  return (
    <section className="max-w-6xl mx-auto py-16 px-4 bg-white">
      <div className="grid grid-cols-1 sm:grid-cols-2 md:grid-cols-3 lg:grid-cols-4 gap-6">
        {products.map((product) => (
          <div key={product._id} className="p-4 border rounded-lg hover:shadow-lg transition duration-300">
            <div className="h-48 w-full relative">
              <Image
                src={product.image_url}
                alt={product.name}
                layout="fill"
                objectFit="cover"
                className="object-center"
              />
            </div>
            <h3 className="text-lg font-medium mb-2">{product.name}</h3>
            <p className="text-sm text-gray-600">Category: {product.category}</p>
            <p className="text-lg font-semibold">£{product.price}</p>
            <p className="text-sm text-gray-500">Quantity: {product.quantity || 'N/A'}</p>
          </div>
        ))}
      </div>
      <div className="text-center mt-10">
        <Link href="/product-listings">
          <button className="px-6 py-3 bg-gray-300 rounded-md hover:bg-gray-800 hover:text-white transition">
            View Collection
          </button>
        </Link>
      </div>
    </section>
  );
};

export default Listing;
```

Data Dynamically displayed on Frontend:



The Poplar suede sofa
Category:
£980
Quantity: 50

Tropical Vibe
Category:
£550
Quantity: 50

Sleek Living
Category:
£300
Quantity: 50

Serene Seat
Category:
£350
Quantity: 50

Conclusion:

By the end of Day03 Alhamdulillah I learned how to Integrate API and How can I migrate API's data to Sanity CMS and how can I display that dynamically to the Frontend.