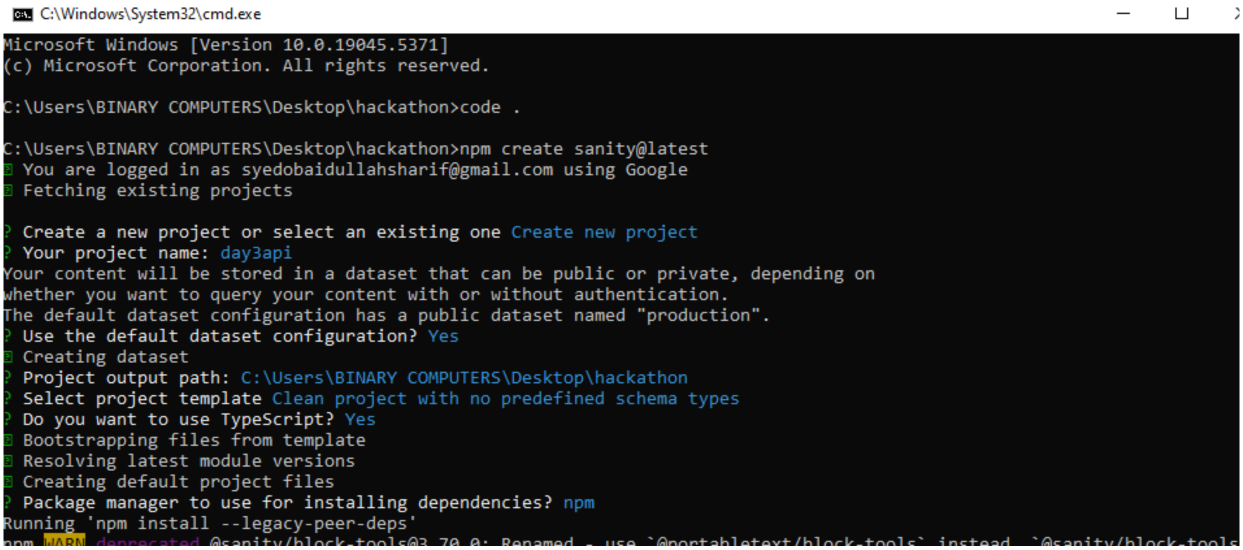
Document Title: “Day 3 API Integration Report”

What to submit 👍
A report documenting:

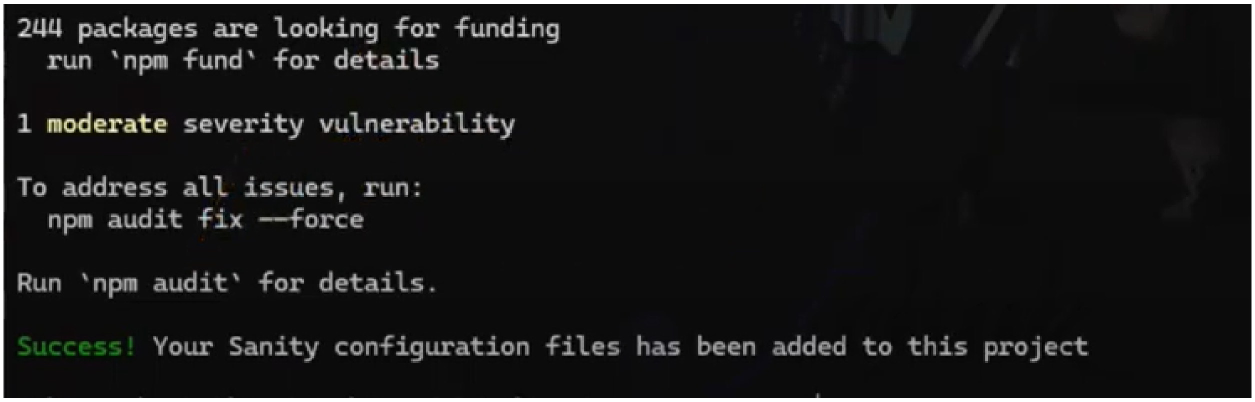
- API integration process.
- Adjustments made to schemas.
- Migration steps and tools used.

Screenshots of:

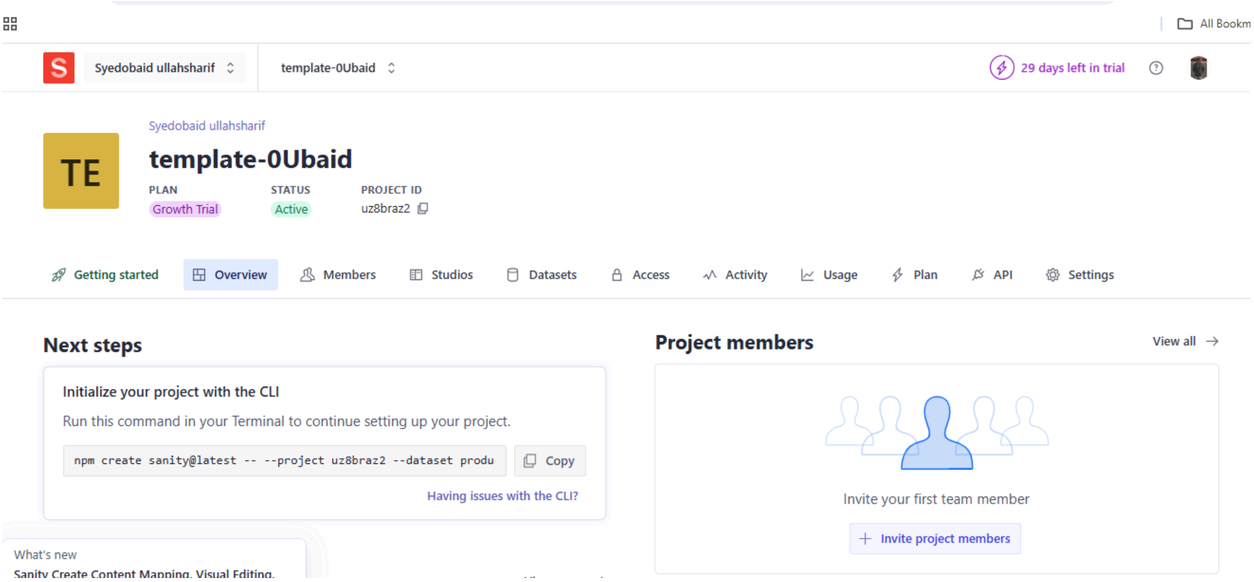
- API calls.
- Data successfully displayed in the frontend.
- Populated Sanity CMS fields.
- Code snippets for API integration and migration scripts.



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

C:\Users\BINARY COMPUTERS\Desktop\hackathon>code .

C:\Users\BINARY COMPUTERS\Desktop\hackathon>npm create sanity@latest
✓ You are logged in as syedobaidullahsharif@gmail.com using Google
✓ Fetching existing projects

? Create a new project or select an existing one Create new project
? Your project name: day3api
Your content will be stored in a dataset that can be public or private, depending on
whether you want to query your content with or without authentication.
The default dataset configuration has a public dataset named "production".
? Use the default dataset configuration? Yes
✓ Creating dataset
? Project output path: C:\Users\BINARY COMPUTERS\Desktop\hackathon
? Select project template Clean project with no predefined schema types
? Do you want to use TypeScript? Yes
✓ Bootstrapping files from template
✓ Resolving latest module versions
✓ Creating default project files
? Package manager to use for installing dependencies? npm
Running 'npm install --legacy-peer-deps'
npm WARN deprecated @sanity/block-tools@3.70.0: Renamed - use `@portabletext/block-tools` instead. `@sanity/block-tools
```



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

1 moderate 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
```



Syedobaid ullahsharif

template-0Ubaid

PLAN Growth Trial

STATUS Active

PROJECT ID uz8braz2

Next steps

Initialize your project with the CLI

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

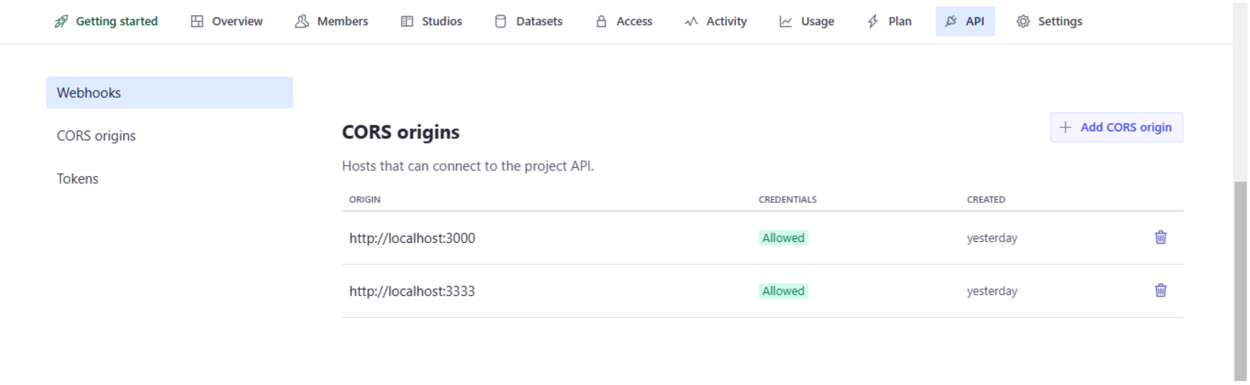
```
npm create sanity@latest -- --project uz8braz2 --dataset produ
```

What's new

Sanity Create Content Mapping, Visual Editing,

Project members

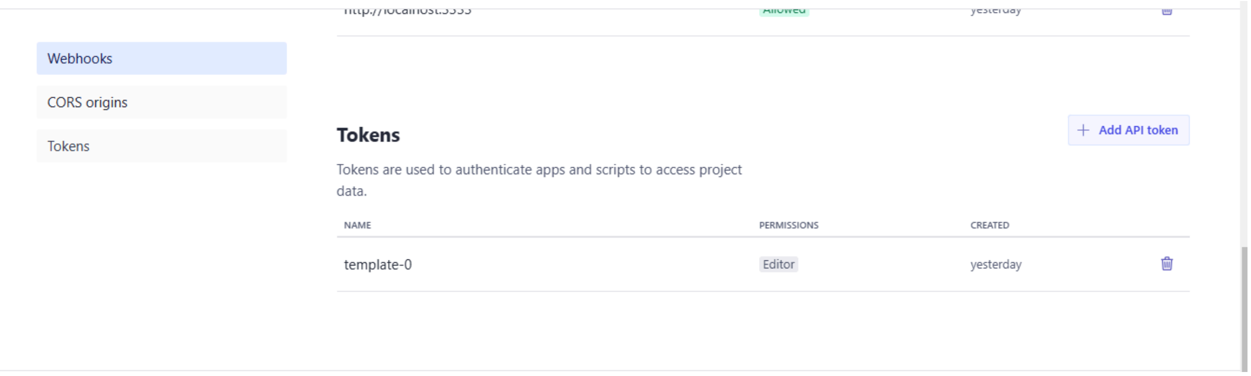
Invite your first team member



CORS origins

Hosts that can connect to the project API.

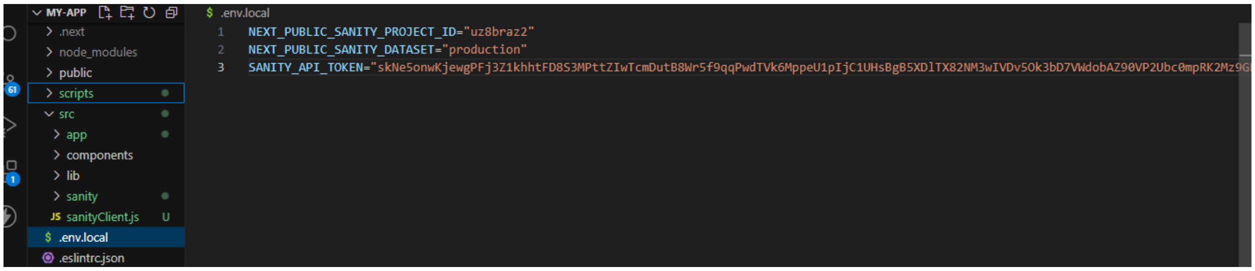
ORIGIN	CREDENTIALS	CREATED
http://localhost:3000	Allowed	yesterday
http://localhost:3333	Allowed	yesterday



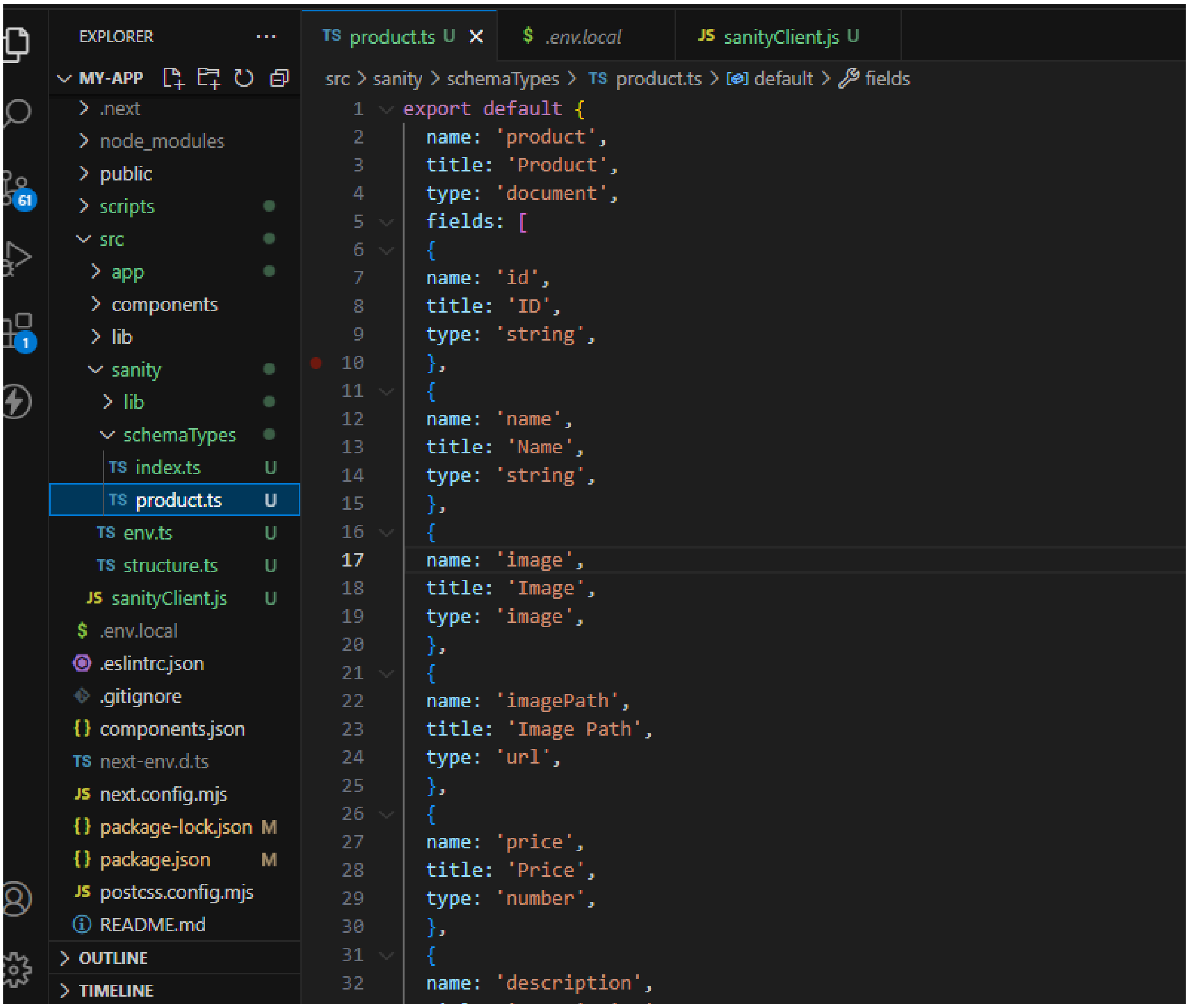
Tokens

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

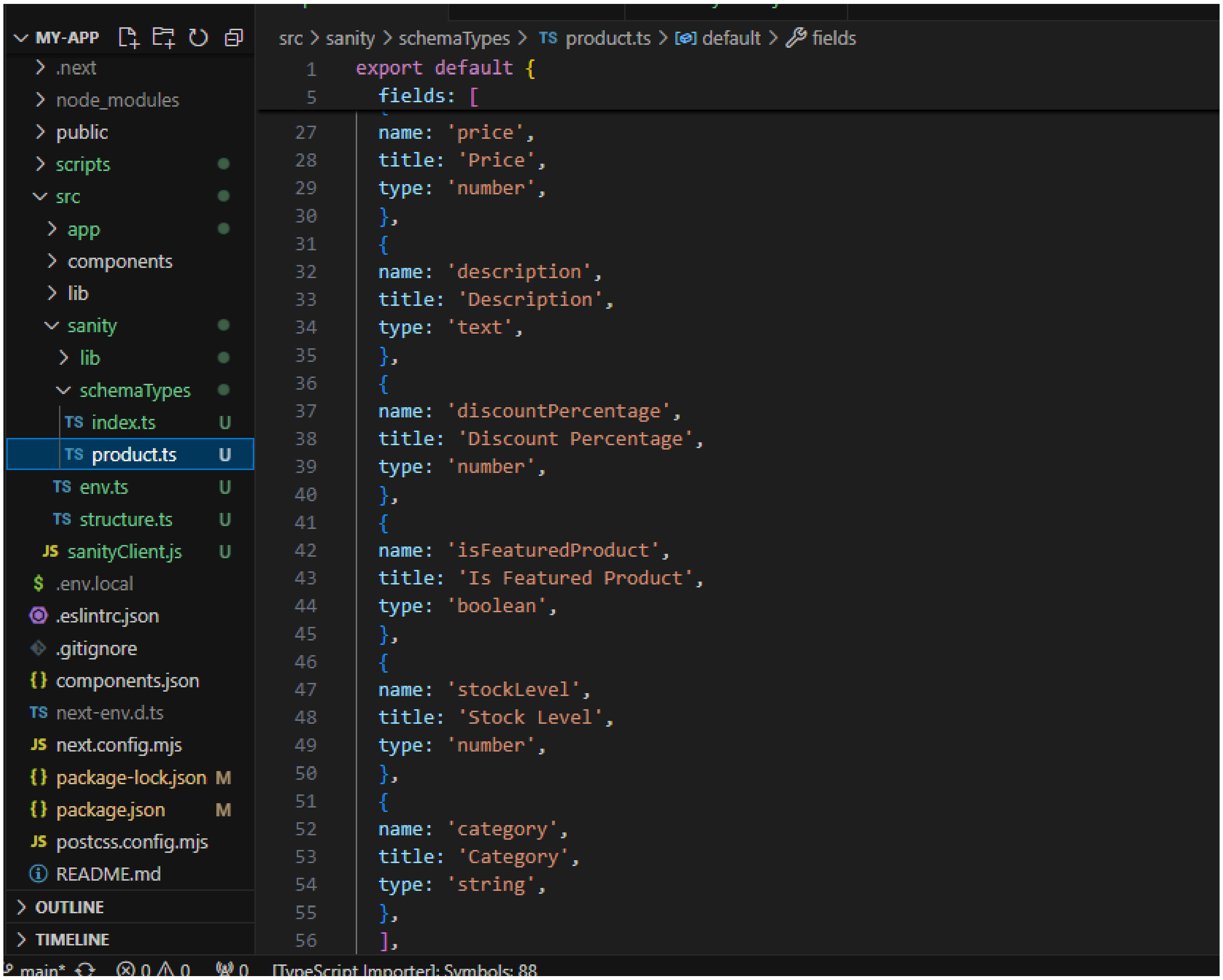
NAME	PERMISSIONS	CREATED
template-0	Editor	yesterday



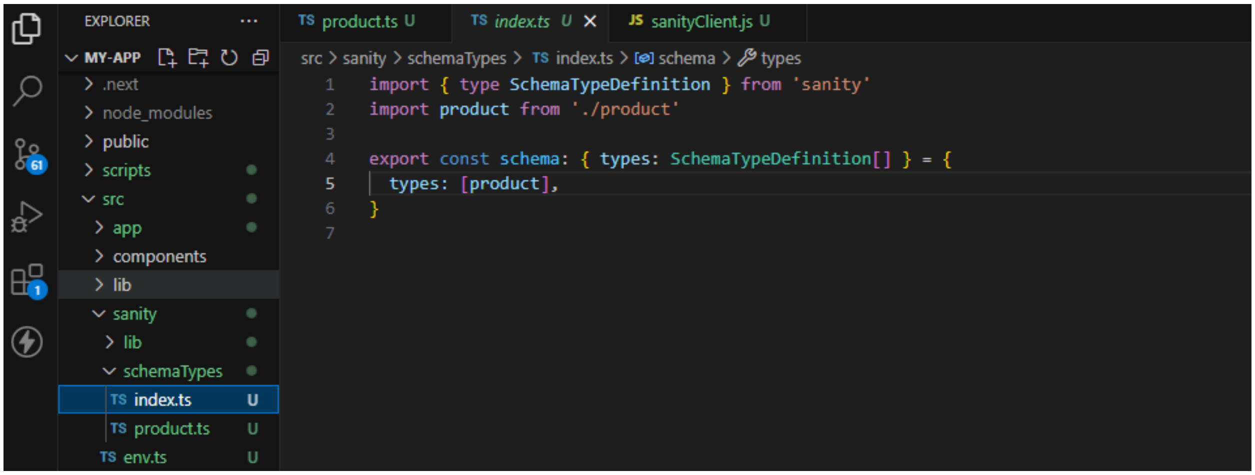
```
NEXT_PUBLIC_SANITY_PROJECT_ID="uz8braz2"
NEXT_PUBLIC_SANITY_DATASET="production"
SANITY_API_TOKEN="skNe5onwKjewgPFj3Z1khhtFD8S3MPttZIwTcmDutB8Wr5f9qqPwdTVk6MppeU1pIjC1UHsBgB5XDlTX82NM3wIVDv5Ok3bD7VWdobAZ90VP2Ubc0mpRK2Mz9G
```



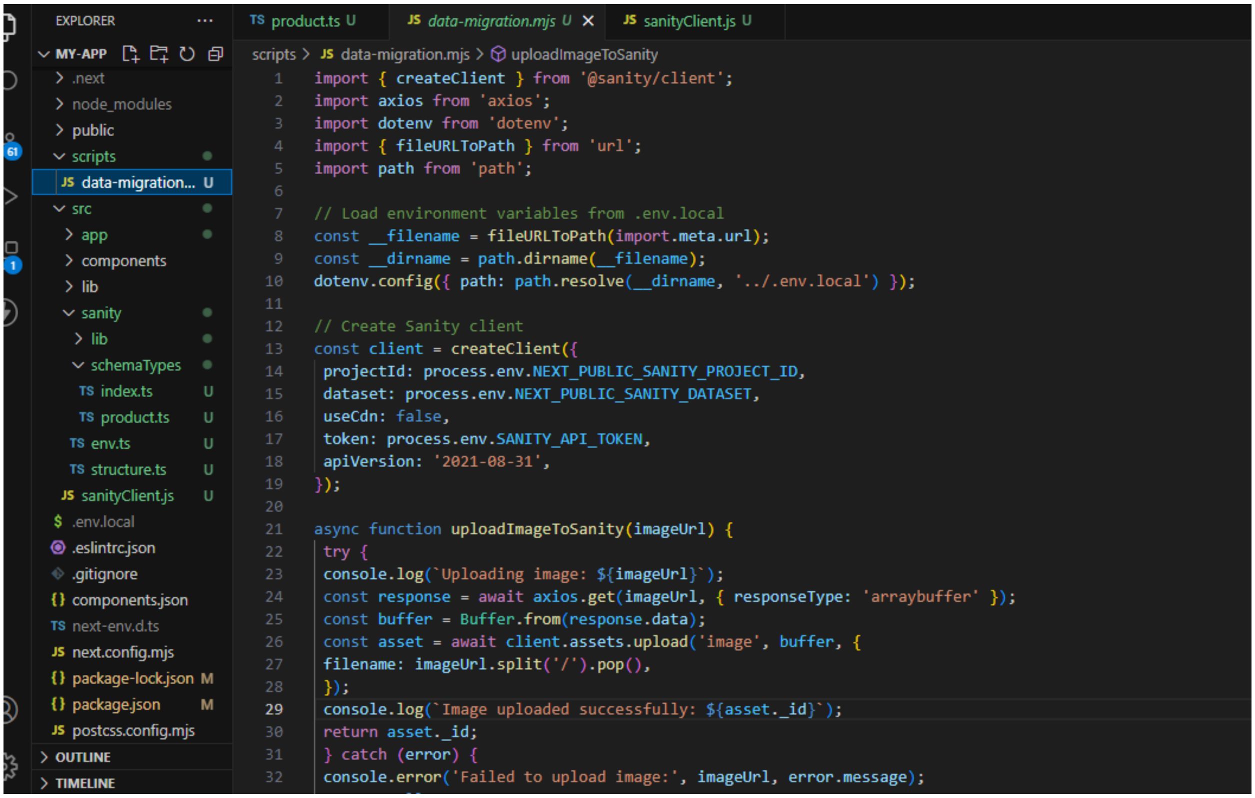
```
export default {
  name: 'product',
  title: 'Product',
  type: 'document',
  fields: [
  {
  name: 'id',
  title: 'ID',
  type: 'string',
  },
  {
  name: 'name',
  title: 'Name',
  type: 'string',
  },
  {
  name: 'image',
  title: 'Image',
  type: 'image',
  },
  {
  name: 'imagePath',
  title: 'Image Path',
  type: 'url',
  },
  {
  name: 'price',
  title: 'Price',
  type: 'number',
  },
  {
  name: 'description',
```



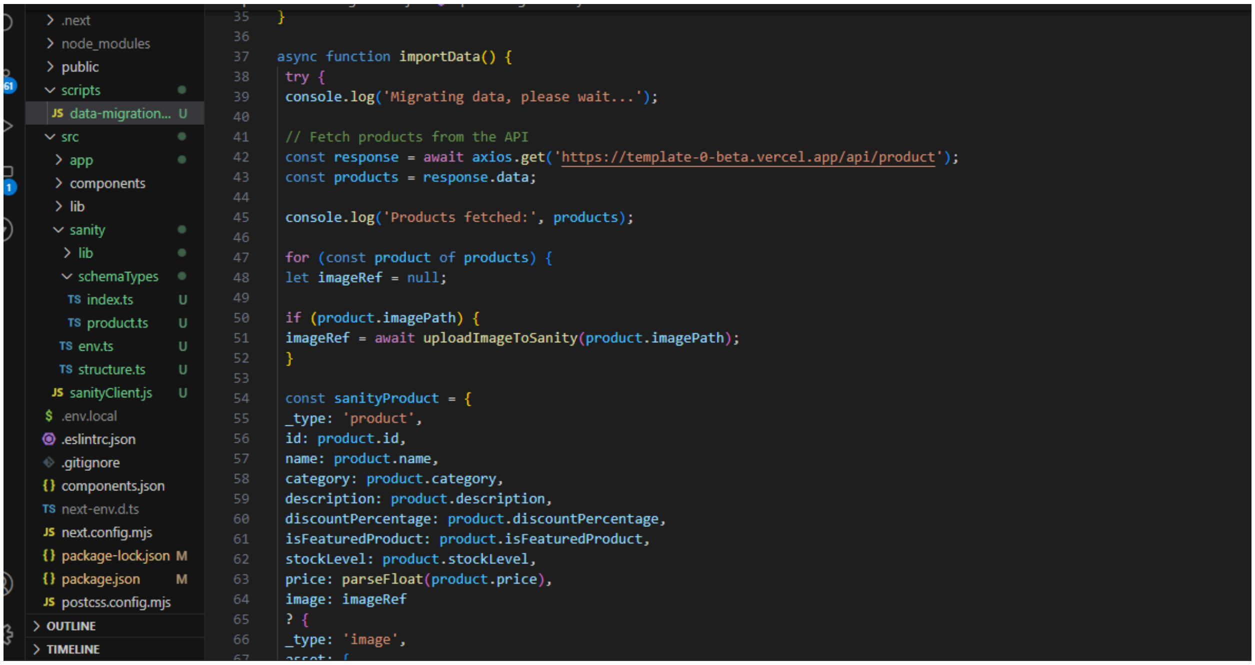
```
export default {
  fields: [
  name: 'price',
  title: 'Price',
  type: 'number',
  },
  {
  name: 'description',
  title: 'Description',
  type: 'text',
  },
  {
  name: 'discountPercentage',
  title: 'Discount Percentage',
  type: 'number',
  },
  {
  name: 'isFeaturedProduct',
  title: 'Is Featured Product',
  type: 'boolean',
  },
  {
  name: 'stockLevel',
  title: 'Stock Level',
  type: 'number',
  },
  {
  name: 'category',
  title: 'Category',
  type: 'string',
  },
  ],
```



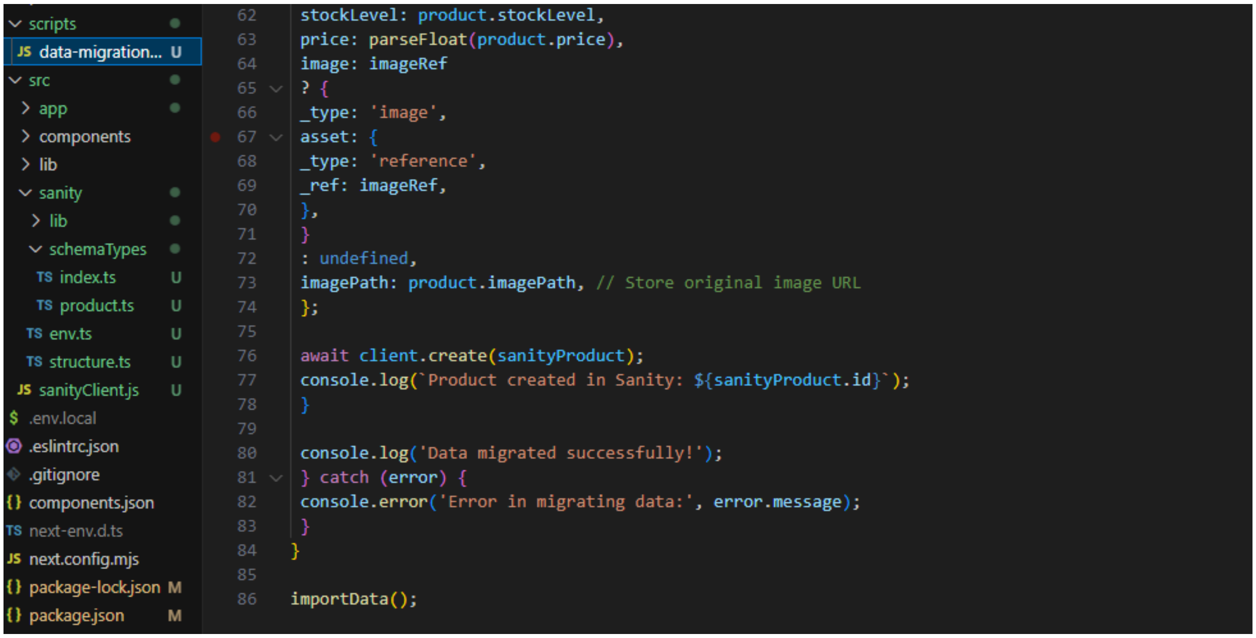
```
import { type SchemaTypeDefinition } from 'sanity'
import product from './product'

export const schema: { types: SchemaTypeDefinition[] } = {
  types: [product],
}
```



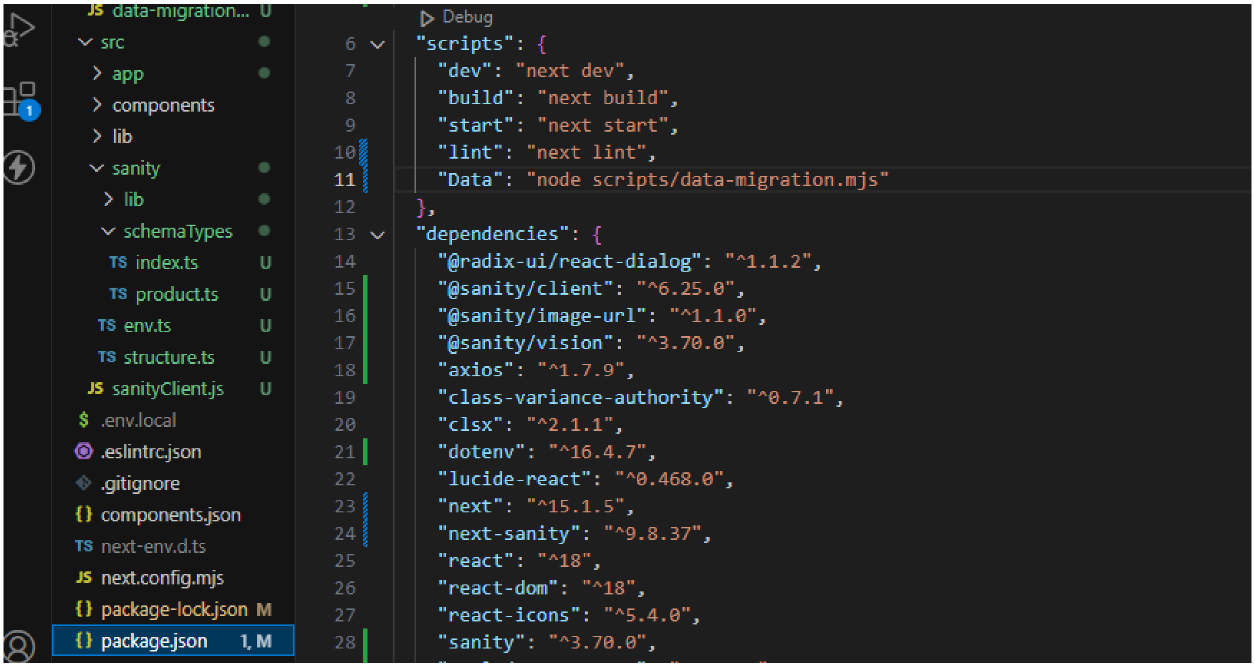
```javascript
import { createClient } from '@sanity/client';
import axios from 'axios';
import dotenv from 'dotenv';
import { fileURLToPath } from 'url';
import path from 'path';

// Load environment variables from .env.local
const __filename = fileURLToPath(import.meta.url);
const __dirname = path.dirname(__filename);
dotenv.config({ path: path.resolve(__dirname, '../.env.local') });

// Create Sanity client
const client = createClient({
 projectId: process.env.NEXT_PUBLIC_SANITY_PROJECT_ID,
 dataset: process.env.NEXT_PUBLIC_SANITY_DATASET,
 useCdn: false,
 token: process.env.SANITY_API_TOKEN,
 apiVersion: '2021-08-31',
});

async function uploadImageToSanity(imageUrl) {
 try {
 console.log(`Uploading image: ${imageUrl}`);
 const response = await axios.get(imageUrl, { responseType: 'arraybuffer' });
 const buffer = Buffer.from(response.data);
 const asset = await client.assets.upload('image', buffer, {
 filename: imageUrl.split('/').pop(),
 });
 console.log(`Image uploaded successfully: ${asset._id}`);
 return asset._id;
 } catch (error) {
 console.error('Failed to upload image:', imageUrl, error.message);
```



```javascript
}

async function importData() {
 try {
 console.log('Migrating data, please wait...');

 // Fetch products from the API
 const response = await axios.get('https://template-0-beta.vercel.app/api/product');
 const products = response.data;

 console.log('Products fetched:', products);

 for (const product of products) {
 let imageRef = null;

 if (product.imagePath) {
 imageRef = await uploadImageToSanity(product.imagePath);
 }

 const sanityProduct = {
 _type: 'product',
 id: product.id,
 name: product.name,
 category: product.category,
 description: product.description,
 discountPercentage: product.discountPercentage,
 isFeaturedProduct: product.isFeaturedProduct,
 stockLevel: product.stockLevel,
 price: parseFloat(product.price),
 image: imageRef
 ? {
 _type: 'image',
 asset: {
```



```
 stockLevel: product.stockLevel,
 price: parseFloat(product.price),
 image: imageRef
 ? {
 _type: 'image',
 asset: {
 _type: 'reference',
 _ref: imageRef,
 },
 }
 : undefined,
 imagePath: product.imagePath, // Store original image URL
 };

 await client.create(sanityProduct);
 console.log(`Product created in Sanity: ${sanityProduct.id}`);
 }

 console.log('Data migrated successfully!');
 } catch (error) {
 console.error('Error in migrating data:', error.message);
 }
}

importData();
```



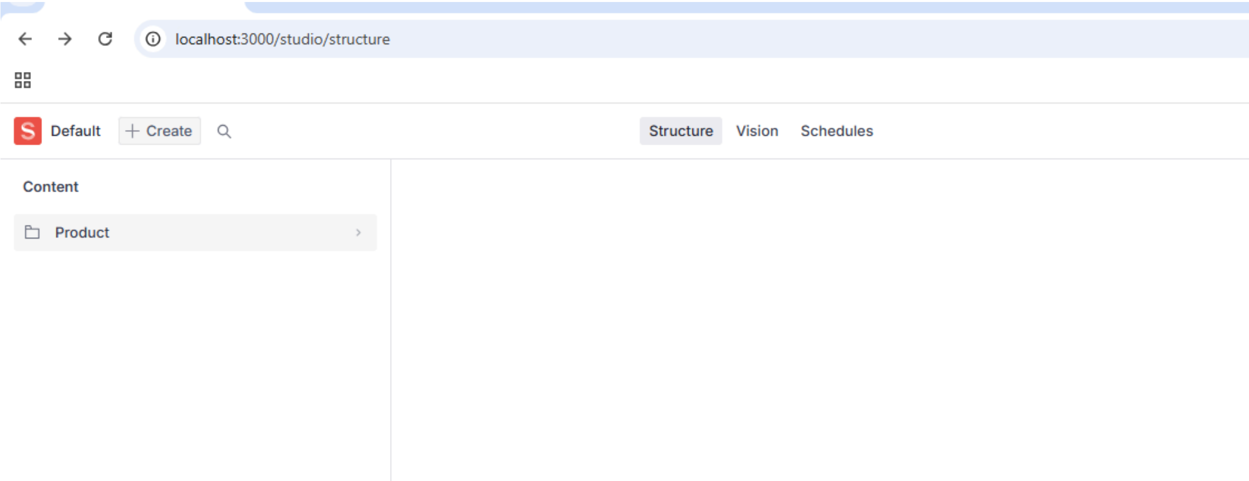
```
"scripts": {
  "dev": "next dev",
  "build": "next build",
  "start": "next start",
  "lint": "next lint",
  "Data": "node scripts/data-migration.mjs"
},
"dependencies": {
  "@radix-ui/react-dialog": "^1.1.2",
  "@sanity/client": "^6.25.0",
  "@sanity/image-url": "^1.1.0",
  "@sanity/vision": "^3.70.0",
  "axios": "^1.7.9",
  "class-variance-authority": "^0.7.1",
  "clsx": "^2.1.1",
  "dotenv": "^16.4.7",
  "lucide-react": "^0.468.0",
  "next": "^15.1.5",
  "next-sanity": "^9.8.37",
  "react": "^18",
  "react-dom": "^18",
  "react-icons": "^5.4.0",
  "sanity": "^3.70.0",
```

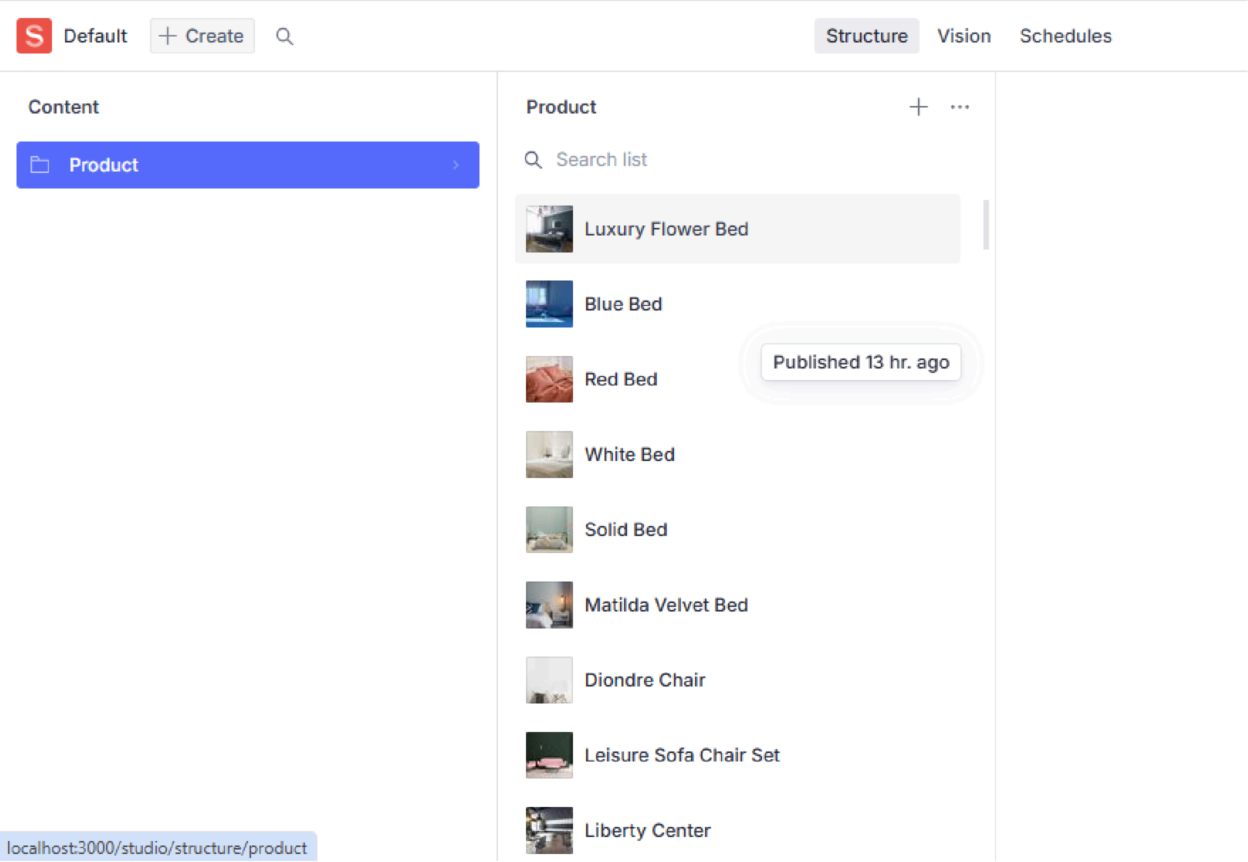
Command run:

```
npm install @sanity axios dotenv
```

Command Run for import Data:

```
Npm run Data
```





Luxury Flower Bed

Blue Bed

Red Bed

Published 13 hr. ago

White Bed

Solid Bed

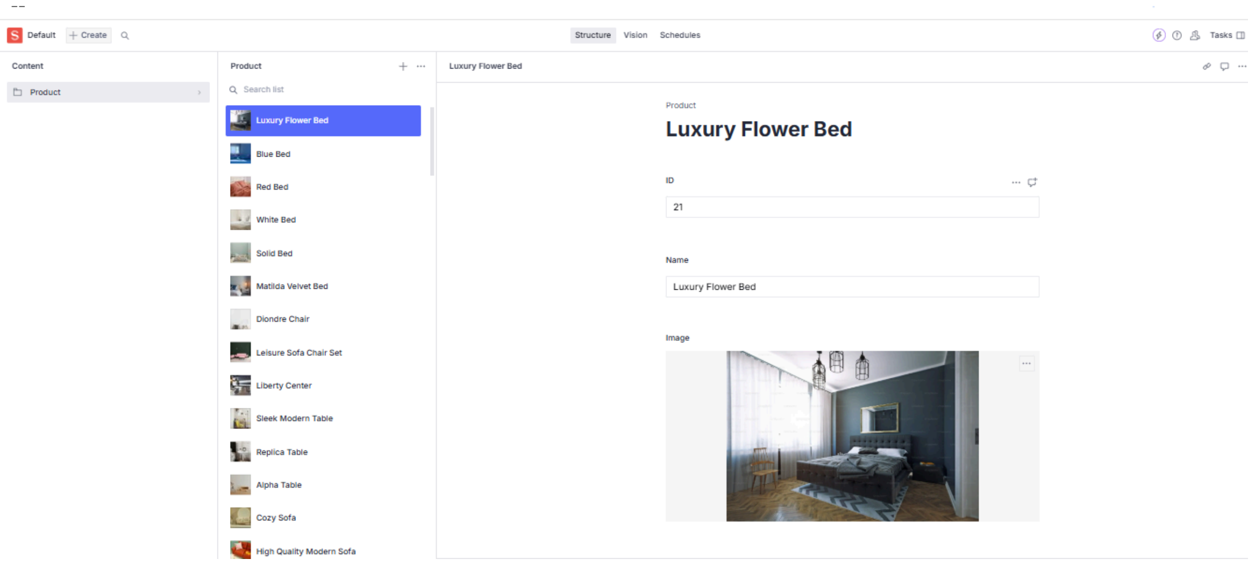
Matilda Velvet Bed

Diondre Chair

Leisure Sofa Chair Set

Liberty Center

localhost:3000/studio/structure/product



Luxury Flower Bed

Blue Bed

Red Bed

White Bed

Solid Bed

Matilda Velvet Bed

Diondre Chair

Leisure Sofa Chair Set

Liberty Center

Sleek Modern Table

Replica Table

Alpha Table

Cozy Sofa

High Quality Modern Sofa

Product

Luxury Flower Bed

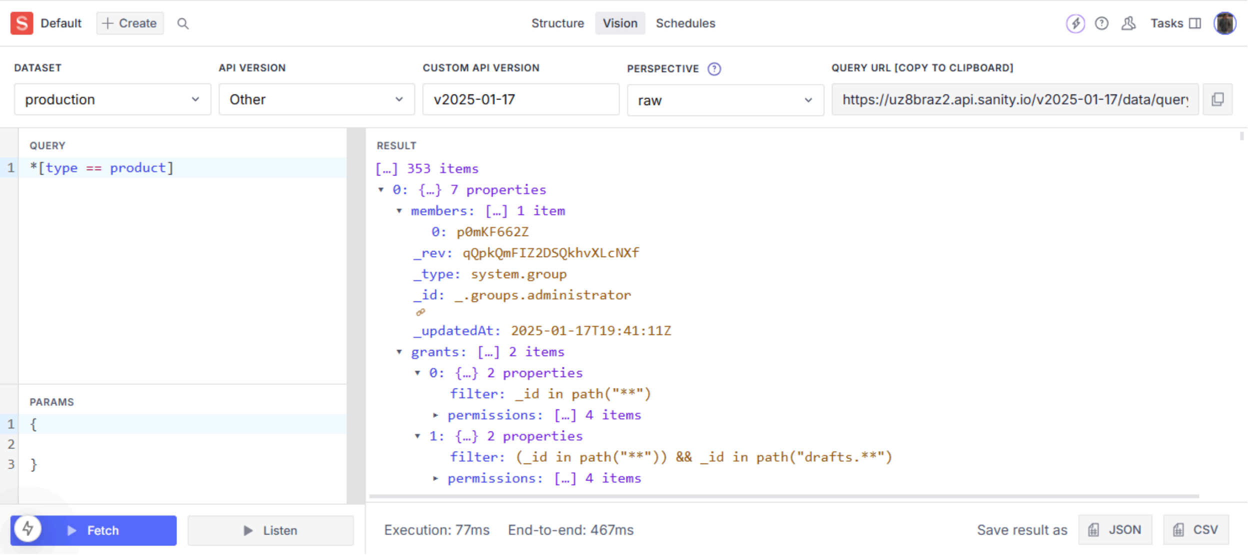
ID

21

Name

Luxury Flower Bed

Image



DATASET production

API VERSION Other

CUSTOM API VERSION v2025-01-17

PERSPECTIVE raw

QUERY URL [COPY TO CLIPBOARD] https://uz8braz2.api.sanity.io/v2025-01-17/data/query

QUERY

```
*[type == product]
```

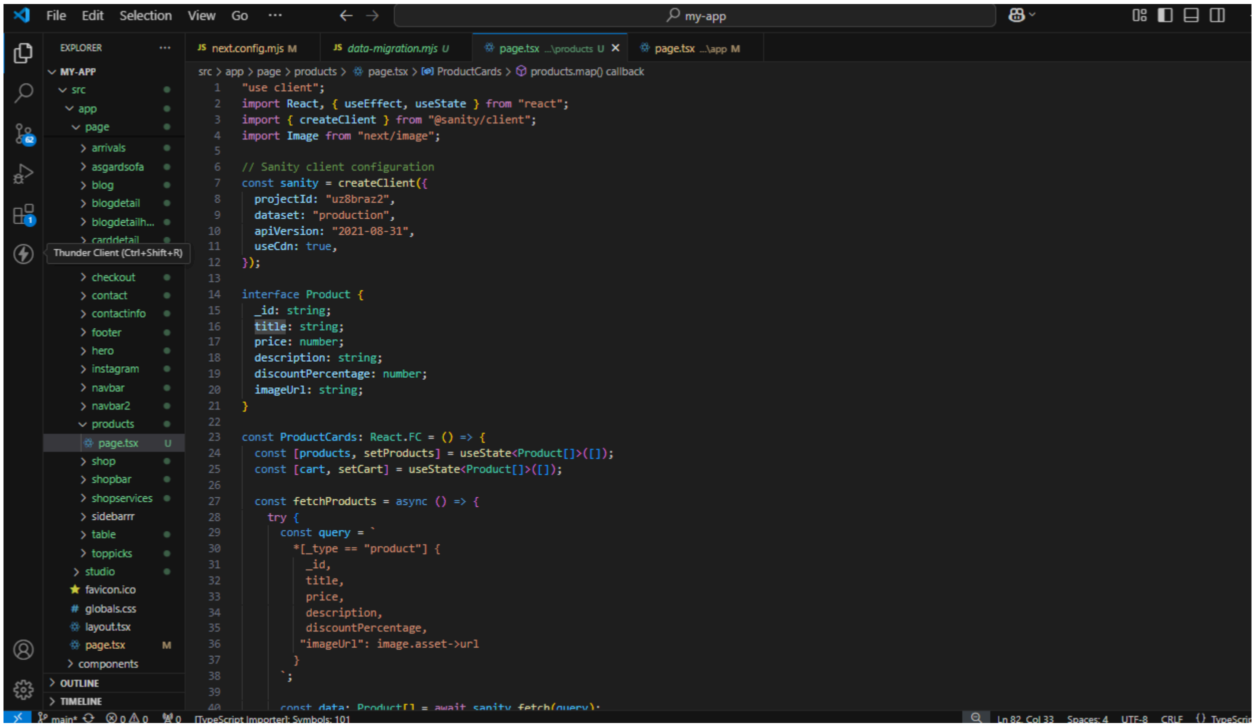
PARAMS

```
{

}
```

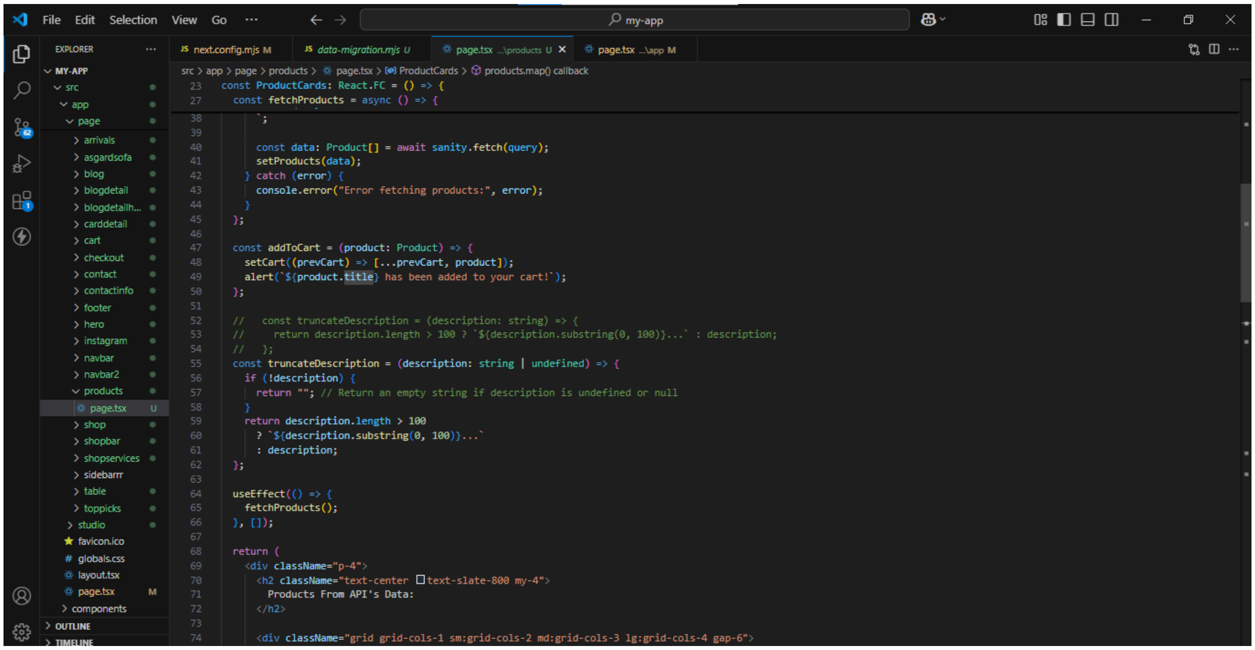
RESULT

```
[…] 353 items
▾ 0: {…} 7 properties
  ▾ members: […] 1 item
      0: p0mKF662Z
    _rev: qQpkQmFIZ2DSQkhvXLcNXf
    _type: system.group
    _id: _.groups.administrator
    _updatedAt: 2025-01-17T19:41:11Z
  ▾ grants: […] 2 items
    ▾ 0: {…} 2 properties
        filter: _id in path("**")
      ▸ permissions: […] 4 items
    ▾ 1: {…} 2 properties
        filter: (_id in path("**")) && _id in path("drafts.**")
      ▸ permissions: […] 4 items
```

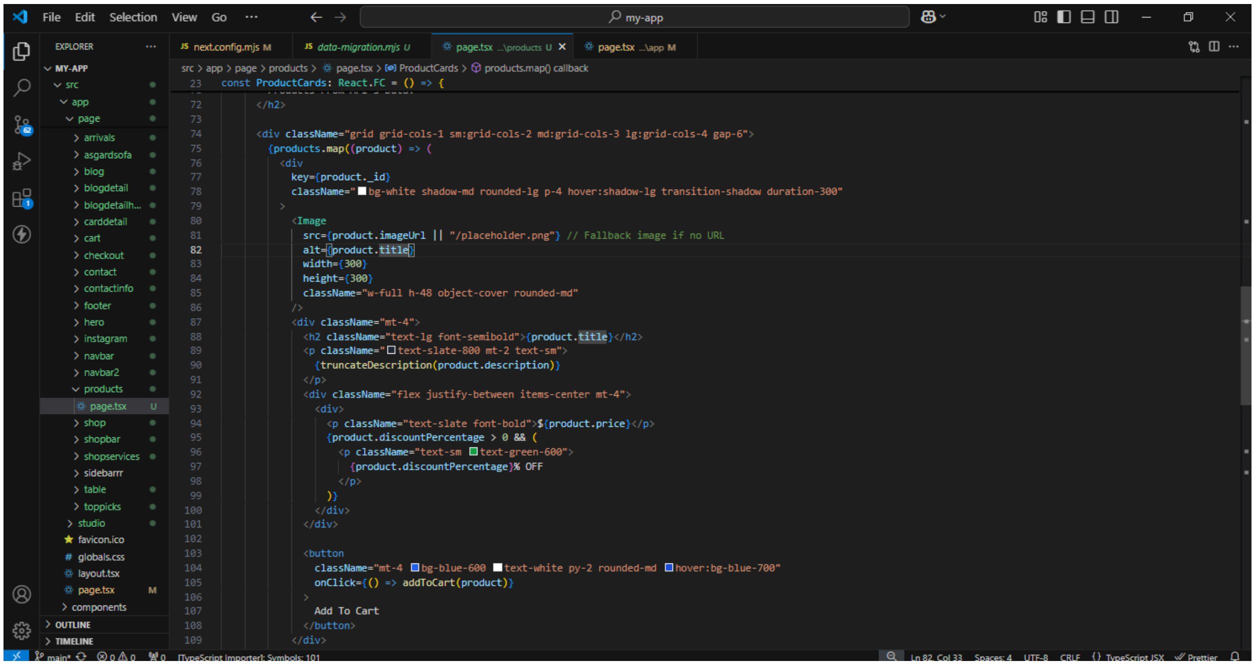
Execution: 77ms End-to-end: 467ms



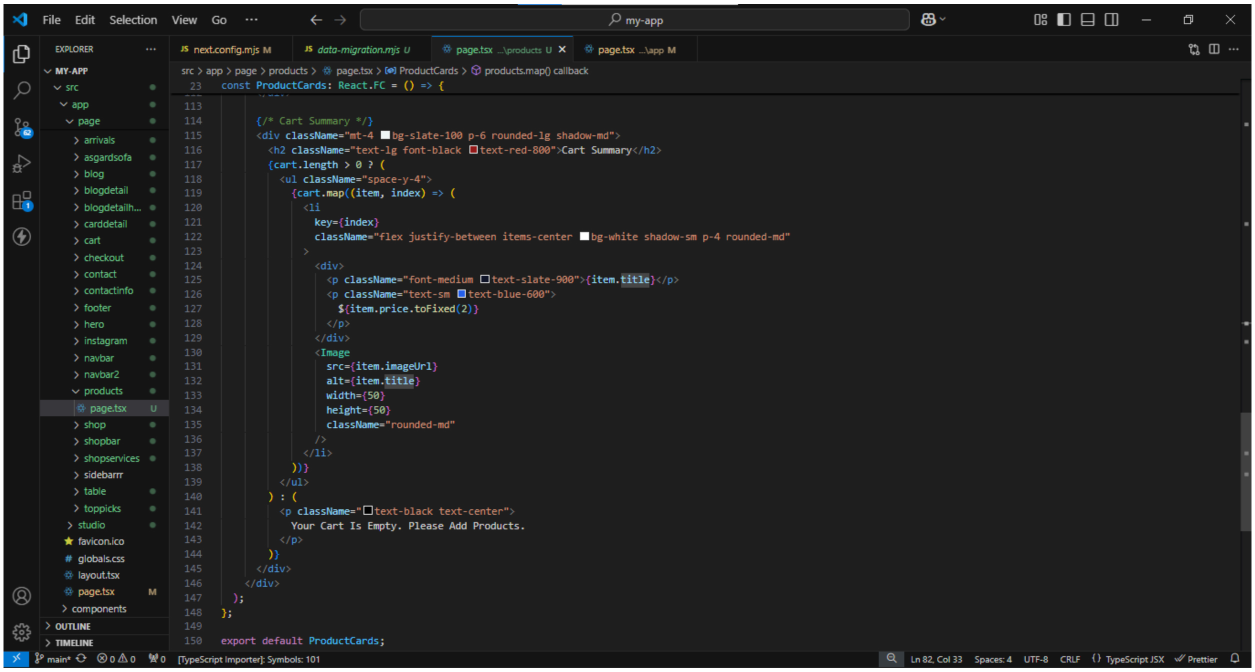
```
"use client";
import React, { useEffect, useState } from "react";
import { createClient } from "@sanity/client";
import Image from "next/image";

// Sanity client configuration
const sanity = createClient({
  projectId: "uz8braz2",
  dataset: "production",
  apiVersion: "2021-08-31",
  useCdn: true,
});

interface Product {
  _id: string;
  title: string;
  price: number;
  description: string;
  discountPercentage: number;
  imageUrl: string;
}

const ProductCards: React.FC = () => {
  const [products, setProducts] = useState<Product[]>([]);
  const [cart, setCart] = useState<Product[]>([]);

  const fetchProducts = async () => {
    try {
      const query = `
        *[_type == "product"] {
          _id,
          title,
          price,
          description,
          discountPercentage,
         "imageUrl": image.asset->url
        }
      `;

      const data: Product[] = await sanity.fetch(query);
```



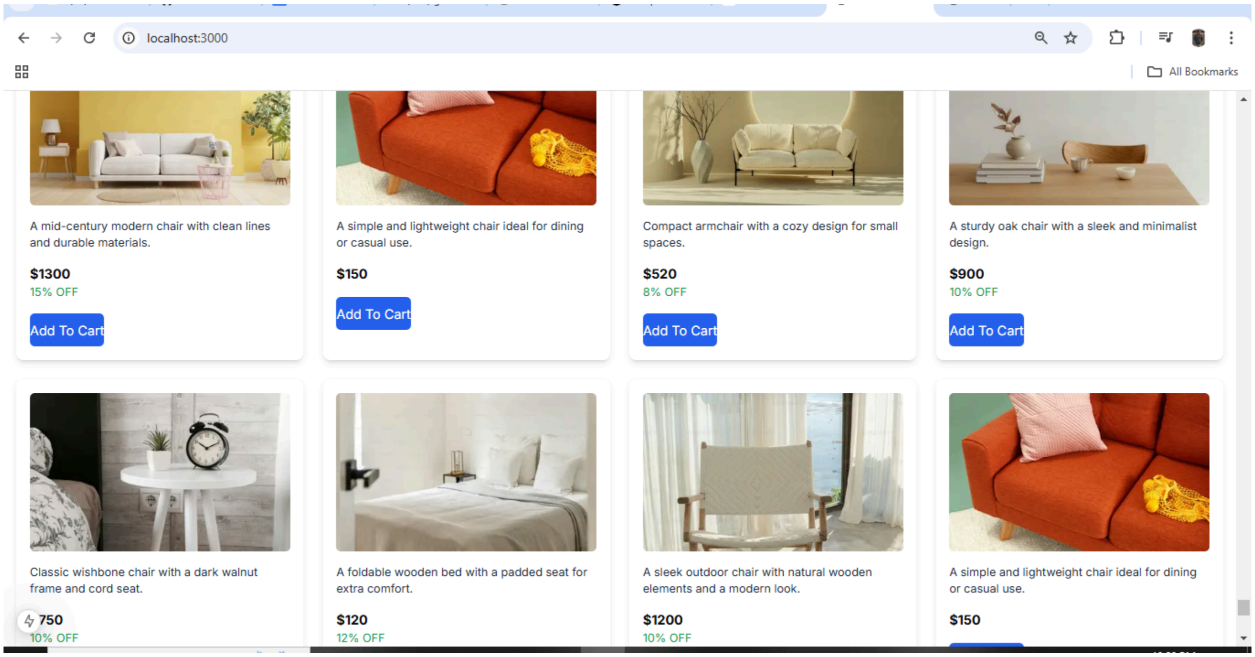
```tsx
const ProductCards: React.FC = () => {
  const fetchProducts = async () => {
      `;

      const data: Product[] = await sanity.fetch(query);
      setProducts(data);
    } catch (error) {
      console.error("Error fetching products:", error);
    }
  };

  const addToCart = (product: Product) => {
    setCart((prevCart) => [...prevCart, product]);
    alert(`${product.title} has been added to your cart!`);
  };

  //   const truncateDescription = (description: string) => {
  //     return description.length > 100 ? `${description.substring(0, 100)}...` : description;
  //   };
  const truncateDescription = (description: string | undefined) => {
    if (!description) {
      return ""; // Return an empty string if description is undefined or null
    }
    return description.length > 100
      ? `${description.substring(0, 100)}...`
      : description;
  };

  useEffect(() => {
    fetchProducts();
  }, []);

  return (
    <div className="p-4">
      <h2 className="text-center text-slate-800 my-4">
        Products From API's Data:
      </h2>

      <div className="grid grid-cols-1 sm:grid-cols-2 md:grid-cols-3 lg:grid-cols-4 gap-6">
```



```tsx
const ProductCards: React.FC = () => {
      </h2>

      <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="bg-white shadow-md rounded-lg p-4 hover:shadow-lg transition-shadow duration-300"
          >
            <Image
              src={product.imageUrl || "/placeholder.png"} // Fallback image if no URL
              alt={product.title}
              width={300}
              height={300}
              className="w-full h-48 object-cover rounded-md"
            />
            <div className="mt-4">
              <h2 className="text-lg font-semibold">{product.title}</h2>
              <p className="text-slate-800 mt-2 text-sm">
                {truncateDescription(product.description)}
              </p>
              <div className="flex justify-between items-center mt-4">
                <div>
                  <p className="text-slate font-bold">${product.price}</p>
                  {product.discountPercentage > 0 && (
                    <p className="text-sm text-green-600">
                      {product.discountPercentage}% OFF
                    </p>
                  )}
                </div>
              </div>

              <button
                className="mt-4 bg-blue-600 text-white py-2 rounded-md hover:bg-blue-700"
                onClick={() => addToCart(product)}
              >
                Add To Cart
              </button>
            </div>
```



```tsx
const ProductCards: React.FC = () => {

      {/* Cart Summary */}
      <div className="mt-4 bg-slate-100 p-6 rounded-lg shadow-md">
        <h2 className="text-lg font-black text-red-800">Cart Summary</h2>
        {cart.length > 0 ? (
          <ul className="space-y-4">
            {cart.map((item, index) => (
              <li
                key={index}
                className="flex justify-between items-center bg-white shadow-sm p-4 rounded-md"
              >
                <div>
                  <p className="font-medium text-slate-900">{item.title}</p>
                  <p className="text-sm text-blue-600">
                    ${item.price.toFixed(2)}
                  </p>
                </div>
                <Image
                  src={item.imageUrl}
                  alt={item.title}
                  width={50}
                  height={50}
                  className="rounded-md"
                />
              </li>
            ))}
          </ul>
        ) : (
          <p className="text-black text-center">
            Your Cart Is Empty. Please Add Products.
          </p>
        )}
      </div>
    </div>
  );
};

export default ProductCards;
```



A mid-century modern chair with clean lines and durable materials.

$1300

15% OFF

Add To Cart

A simple and lightweight chair ideal for dining or casual use.

$150

Add To Cart

Compact armchair with a cozy design for small spaces.

$520

8% OFF

Add To Cart

A sturdy oak chair with a sleek and minimalist design.

$900

10% OFF

Add To Cart

Classic wishbone chair with a dark walnut frame and cord seat.

750

10% OFF

A foldable wooden bed with a padded seat for extra comfort.

$120

12% OFF

A sleek outdoor chair with natural wooden elements and a modern look.

$1200

10% OFF

A simple and lightweight chair ideal for dining or casual use.

$150