

# "Day 3 - API Integration Report - [Avion E-commerce]"

o A report documenting:

□ API integration process.

Project Setup:

Sanity installation:

```
C:\Windows\System32\cmd.exe
Need to install the following packages:
create-sanity@3.72.1
Ok to proceed? (y) y

> e-commerce-hackathon-template-2@0.1.0 npx
> create-sanity

You are logged in as rababmuhammadhussain@gmail.com using Google
Fetching existing projects

? Create a new project or select an existing one Create new project
? Your project name: day3-api-migration
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
? 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

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

API:

Gmail

YouTube

Maps

Download

All Bookmarks

Srabab muhammad

day3-api-migration

30 days left in trial

DA

rabab muhammad

day3-api-migration

PLAN

Growth Trial

STATUS

Active

PROJECT ID

uz5dxxrb

Getting started

Overview

Members

Studios

Datasets

Access

Activity

Usage

Plan

API

Settings

Webhooks

CORS origins

Tokens

GROQ-powered webhooks

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

Learn more about webhooks

+ Create webhook

0 of 2 webhooks (2 included in plan)

Get more webhooks

## CORS origin:

Gmail

YouTube

Maps

Download

All Bookmarks

Srabab muhammad

day3-api-migration

30 days left in trial

Getting started

Overview

Members

Studios

Datasets

Access

Activity

Usage

Plan

API

Settings

Webhooks

CORS origins

Tokens

CORS origins

Hosts that can connect to the project API.

+ Add CORS origin

ORIGIN	CREDENTIALS	CREATED	
http://localhost:3000	Allowed	47 minutes	
http://localhost:3333	Allowed	47 minutes	

## Add API Token:

Gmail

YouTube

Maps

Download

All Bookmarks

Srabab muhammad

day3-api-migration

30 days left in trial

Getting started

Overview

Members

Studios

Datasets

Access

Activity

Usage

Plan

API

Settings

Webhooks

CORS origins

Tokens

Name

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

day-3

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

☒ Read and write access to all datasets, with access to project settings for developers. (Tokens: read+write)

Editor

☐ Read and write access to all datasets, with limited access to project settings. (Tokens: read+write)

Viewer

☐ Read access to all datasets, with limited access to project settings. (Tokens: read-only)

Save

Cancel

What's new

Sanity Create Content Mapping, Visual Editing, and Content Releases

S

rabab muhammad

day3-api-migration

30 days left in trial

Getting started

Overview

Members

Studios

Datasets

Access

Activity

Usage

Plan

API

Settings

Webhooks

CORS origins

Tokens

Tokens

+ Add API token

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

NAME	PERMISSIONS	CREATED
day-3	Developer	0 seconds

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

skx9XBmEP4ubvrhEaof22H7NBly1YspIaUr0ZVvx1YsqJBLqS1fCz2N5gpUtS12P2AtHygFwHIjqtIQRyzQDnuPbrtIi9nt9lDd6JtLJF0dXOCLmemWm0jR6f2mz03l3bbwa1NSAwsLmCPNUFBP9DLdRixc5ZGZkuX7pG0XexY6qV2pIr3sH

## Adjustments made to schemas.

### Product Schema:

File Edit Selection View Go Run ...

hackathon-3-template-2

EXPLORER

HACKATHON-3-TEMPLA...

node\_modules

public

src

app

components

lib

sanity

lib

schemaTypes

TS category.ts

TS index.ts

TS product.ts

TS env.ts

TS structure.ts

.env.local

.eslintrc.json

.gitignore

components.json

next.config.mjs

package-lock.json

package.json

postcss.config.mjs

README.md

sanity.cli.ts

sanity.config.ts

OUTLINE

TIMELINE

IDENTIFY

.env.local

TS product.ts

TS category.ts

TS index.ts

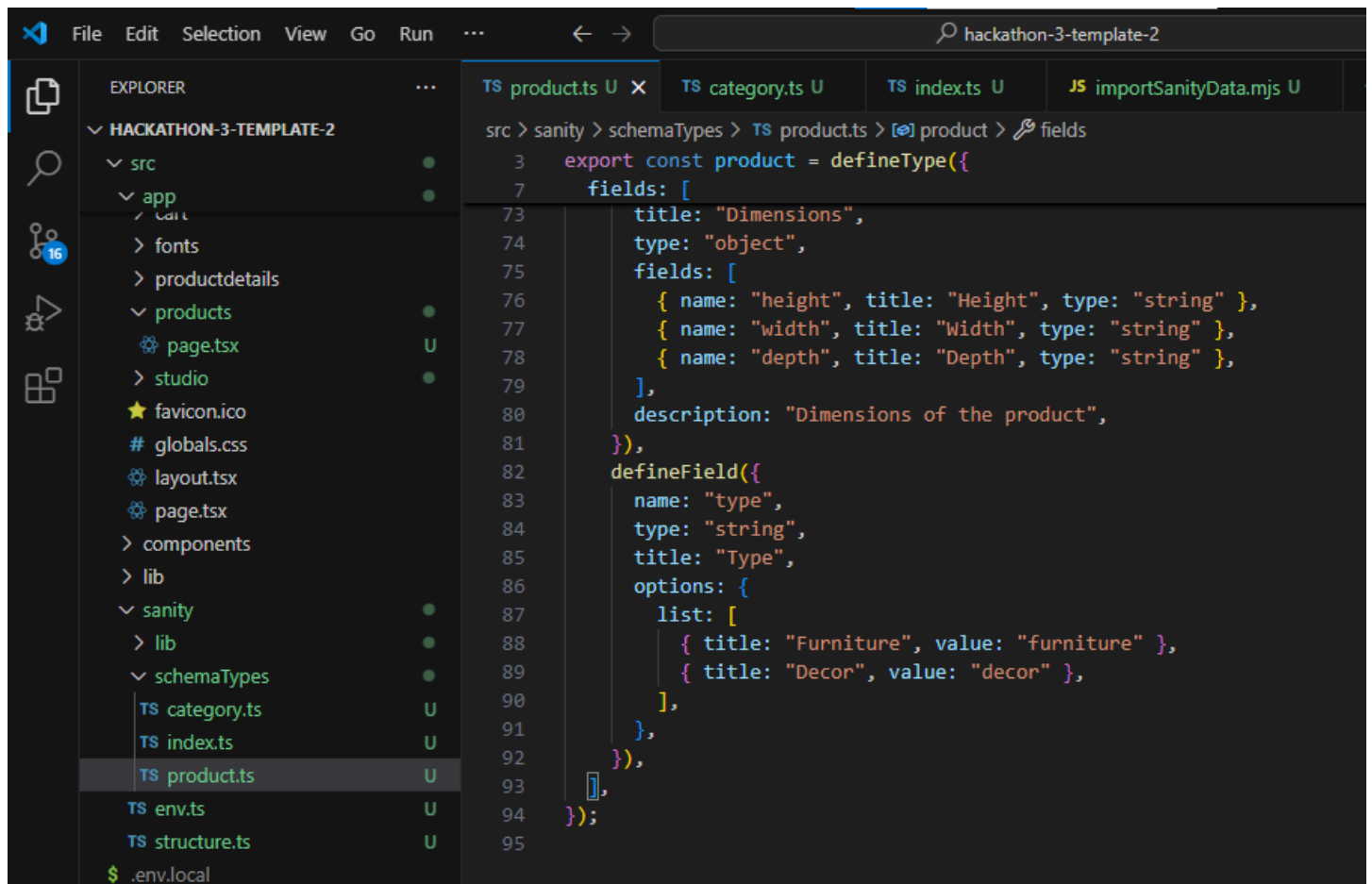
src > sanity > schemaTypes > TS product.ts > product

```
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    defineField({
29      name: "image",
30      type: "image",
31      validation: (rule) => rule.required(),
32
```

The screenshot shows the VS Code editor interface with the Explorer sidebar on the left and the Editor pane on the right. The Explorer sidebar shows the project structure for 'HACKATHON-3-TEMPLA...', with the file 'TS product.ts' selected under the 'src > sanity > schemaTypes' directory. The Editor pane displays the content of 'product.ts', which defines a Sanity schema for a product. The schema includes fields for 'image', 'price', 'quantity', 'tags', and 'description', each with specific types and validation rules.

```
src > sanity > schemaTypes > TS product.ts > [0] product
3  export const product = defineType({
7    fields: [
29      defineField({
30        name: "image",
31        type: "image",
32        validation: (rule) => rule.required(),
33        title: "Product Image"
34      }),
35      defineField({
36        name: "price",
37        type: "number",
38        validation: (rule) => rule.required(),
39        title: "Price",
40      }),
41      defineField({
42        name: "quantity",
43        title: "Quantity",
44        type: "number",
45        validation: (rule) => rule.min(0),
46      }),
47      defineField({
48        name: "tags",
49        type: "array",
50        title: "Tags",
51        of: [{
52          type: "string"
53        }]
54      }),
55      defineField({
56        name: 'description',
57        title: 'Description',
58        type: 'text',
```

## Updated Product Schema:



The screenshot shows the VS Code editor interface. The Explorer sidebar on the left displays the project structure for 'HACKATHON-3-TEMPLATE-2'. The file 'TS product.ts' is selected and highlighted. The main editor pane shows the content of 'TS product.ts' with the following code:

```
src > sanity > schemaTypes > TS product.ts > product > fields
3  export const product = defineType({
7    fields: [
73      title: "Dimensions",
74      type: "object",
75      fields: [
76        { name: "height", title: "Height", type: "string" },
77        { name: "width", title: "Width", type: "string" },
78        { name: "depth", title: "Depth", type: "string" },
79      ],
80      description: "Dimensions of the product",
81    },
82    defineField({
83      name: "type",
84      type: "string",
85      title: "Type",
86      options: {
87        list: [
88          { title: "Furniture", value: "furniture" },
89          { title: "Decor", value: "decor" },
90        ],
91      },
92    }),
93  ],
94 });
95
```

## Category Schema:

hackathon-3-template-2

EXPLORER

- HACKATHON-3-TEMPLA...
  - node\_modules
  - public
  - src
    - app
    - components
    - lib
    - sanity
      - lib
      - schemaTypes
        - TS category.ts U
        - TS index.ts U
        - TS product.ts U
      - env.ts U
      - structure.ts U
    - .env.local
    - .eslintrc.json
    - .gitignore
    - components.json
    - next.config.mjs
    - package-lock.json M
    - package.json M
    - postcss.config.mjs
    - README.md
    - sanity.cli.ts U
    - sanity.config.ts U

src > sanity > schemaTypes > TS category.ts > Category

```
1 import { defineType, defineField } from "sanity";
2
3 export const Category = defineType({
4   name: "category",
5   title: "Category",
6   type: "document",
7   fields: [
8     defineField({
9       name: "name",
10      title: "Name",
11      type: "string",
12      validation: (rule) => rule.required(),
13    }),
14     defineField({
15       name: "slug",
16       title: "Slug",
17       type: "slug",
18       validation: (rule) => rule.required(),
19       options: {
20         source: "name",
21       }
22     })
23   ]
24 })
```

hackathon-3-template-2

EXPLORER

- HACKATHON-3-TEMPLA...
  - public
  - scripts
    - importSanityData.mjs U
  - src
    - app
    - components
    - lib
    - sanity
      - lib
      - schemaTypes
        - TS category.ts U
        - TS index.ts U
        - TS product.ts U

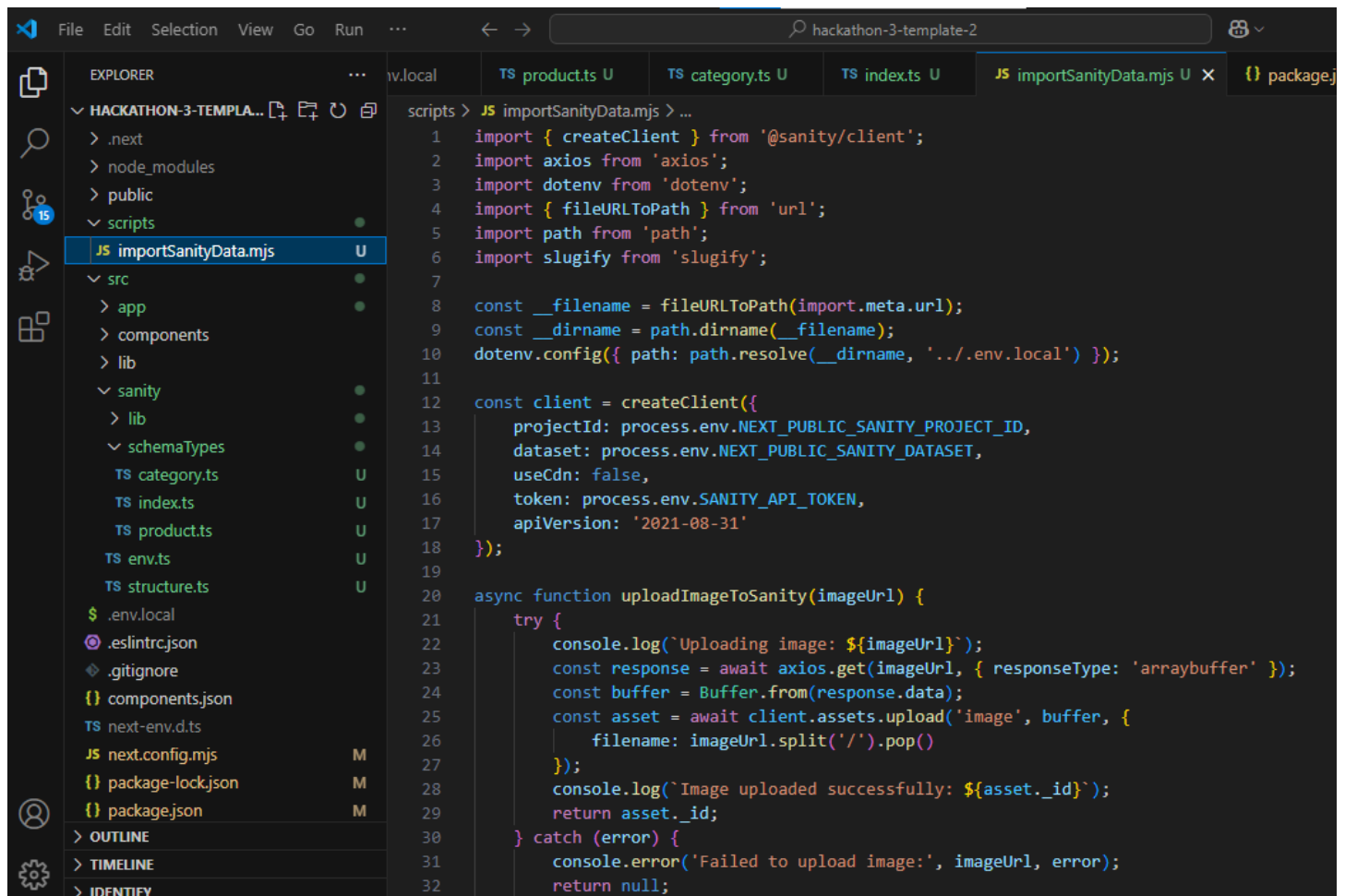
src > sanity > schemaTypes > TS index.ts > schema > types

```
1 import { type SchemaTypeDefinition } from 'sanity'
2 import { product } from './product'
3 import { Category } from './category'
4
5 export const schema: { types: SchemaTypeDefinition[] } = {
6   types: [product, Category],
7 }
8
```

## □ Migration steps and tools used.

- Used a migration script to update existing product documents.
- Tools Used: Sanity CLI, Sanity Client, Node.js

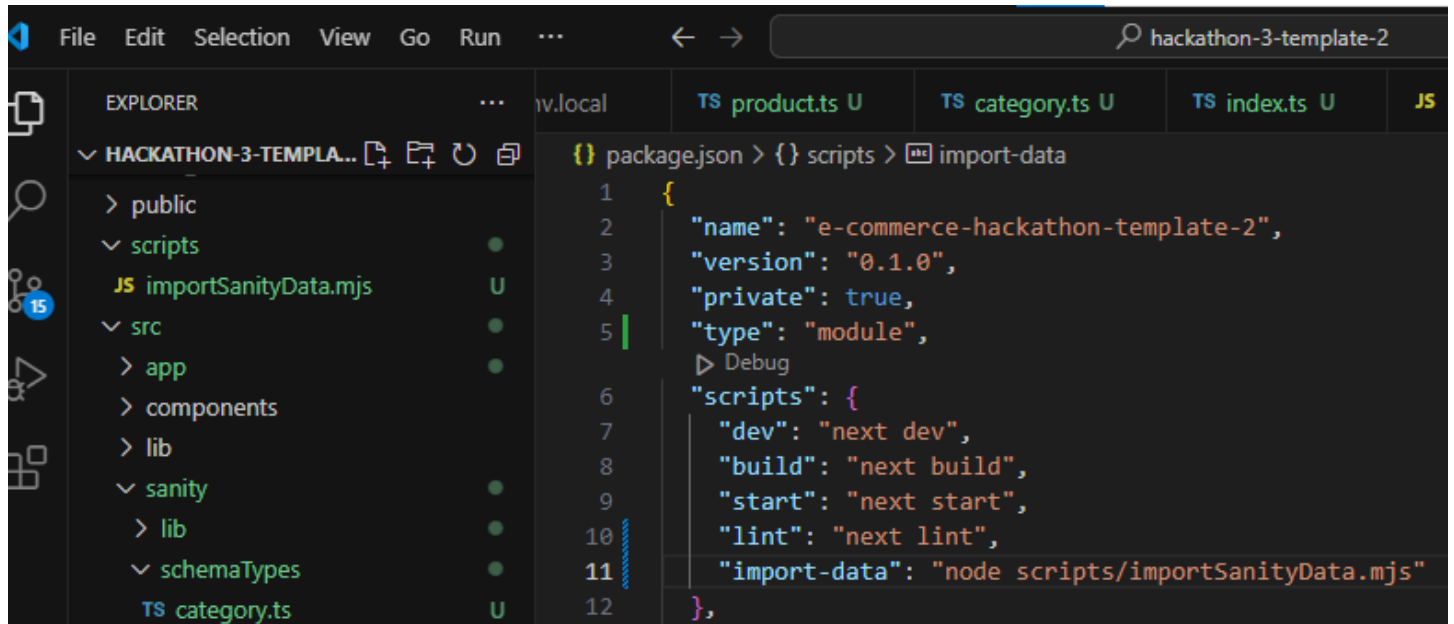
### Migration Script:



The screenshot shows a VS Code editor window with a project named 'hackathon-3-template-2'. The Explorer sidebar on the left shows the file structure, with the 'scripts' folder expanded and 'importSanityData.mjs' selected. The main editor area displays the content of this file, which is a JavaScript migration script. The script imports necessary modules like 'createClient' from '@sanity/client', 'axios', 'dotenv', 'fileURLToPath', 'path', and 'slugify'. It then configures the Sanity client with environment variables for project ID, dataset, token, and API version. Finally, it defines an asynchronous function 'uploadImageToSanity' that takes an 'imageUrl' and attempts to upload it to Sanity's assets, logging the result or any errors.

```
1 import { createClient } from '@sanity/client';
2 import axios from 'axios';
3 import dotenv from 'dotenv';
4 import { fileURLToPath } from 'url';
5 import path from 'path';
6 import slugify from 'slugify';
7
8 const __filename = fileURLToPath(import.meta.url);
9 const __dirname = path.dirname(__filename);
10 dotenv.config({ path: path.resolve(__dirname, '../.env.local') });
11
12 const client = createClient({
13   projectId: process.env.NEXT_PUBLIC_SANITY_PROJECT_ID,
14   dataset: process.env.NEXT_PUBLIC_SANITY_DATASET,
15   useCdn: false,
16   token: process.env.SANITY_API_TOKEN,
17   apiVersion: '2021-08-31'
18 });
19
20 async function uploadImageToSanity(imageUrl) {
21   try {
22     console.log(`Uploading image: ${imageUrl}`);
23     const response = await axios.get(imageUrl, { responseType: 'arraybuffer' });
24     const buffer = Buffer.from(response.data);
25     const asset = await client.assets.upload('image', buffer, {
26       filename: imageUrl.split('/').pop()
27     });
28     console.log(`Image uploaded successfully: ${asset._id}`);
29     return asset._id;
30   } catch (error) {
31     console.error('Failed to upload image:', imageUrl, error);
32     return null;
33   }
34 }
```

### Requirement in Package.json:



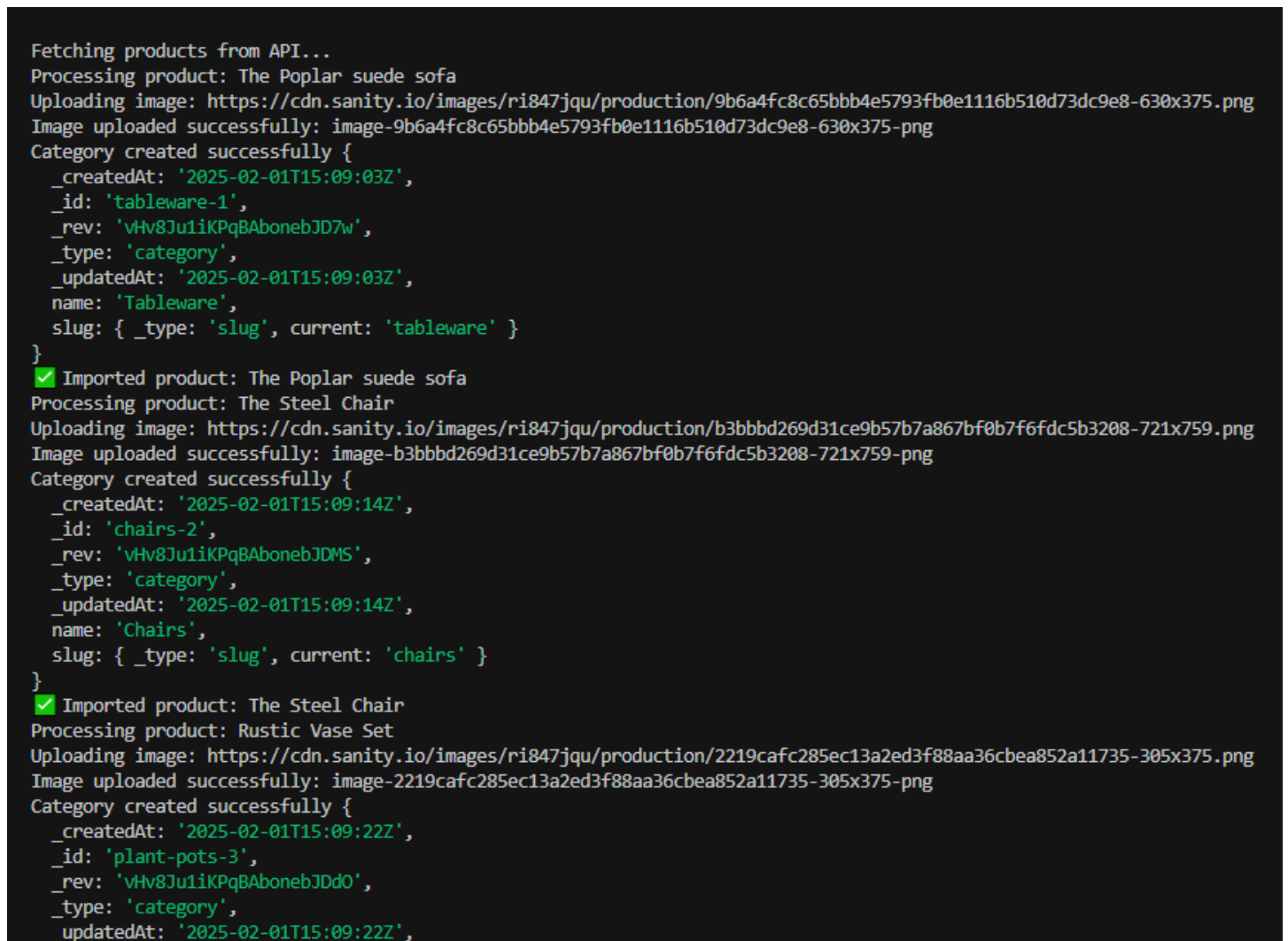
```
File Edit Selection View Go Run ... ← → hackathon-3-template-2

EXPLORER
HACKATHON-3-TEMPLA...
  public
  scripts
  JS importSanityData.mjs
  src
  app
  components
  lib
  sanity
  lib
  schemaTypes
  TS category.ts

package.json
1 {
2   "name": "e-commerce-hackathon-template-2",
3   "version": "0.1.0",
4   "private": true,
5   "type": "module",
6   "scripts": {
7     "dev": "next dev",
8     "build": "next build",
9     "start": "next start",
10    "lint": "next lint",
11    "import-data": "node scripts/importSanityData.mjs"
12  },
```

## □ API calls

### Fetching Data from API to Sanity:



```
Fetching products from API...
Processing product: The Poplar suede sofa
Uploading image: https://cdn.sanity.io/images/ri847jqu/production/9b6a4fc8c65bbb4e5793fb0e1116b510d73dc9e8-630x375.png
Image uploaded successfully: image-9b6a4fc8c65bbb4e5793fb0e1116b510d73dc9e8-630x375-png
Category created successfully {
  _createdAt: '2025-02-01T15:09:03Z',
  _id: 'tableware-1',
  _rev: 'vHv8Ju1iKPqBAbonebJD7w',
  _type: 'category',
  _updatedAt: '2025-02-01T15:09:03Z',
  name: 'Tableware',
  slug: { _type: 'slug', current: 'tableware' }
}
✅ Imported product: The Poplar suede sofa
Processing product: The Steel Chair
Uploading image: https://cdn.sanity.io/images/ri847jqu/production/b3bbbd269d31ce9b57b7a867bf0b7f6fdc5b3208-721x759.png
Image uploaded successfully: image-b3bbbd269d31ce9b57b7a867bf0b7f6fdc5b3208-721x759-png
Category created successfully {
  _createdAt: '2025-02-01T15:09:14Z',
  _id: 'chairs-2',
  _rev: 'vHv8Ju1iKPqBAbonebJDMS',
  _type: 'category',
  _updatedAt: '2025-02-01T15:09:14Z',
  name: 'Chairs',
  slug: { _type: 'slug', current: 'chairs' }
}
✅ Imported product: The Steel Chair
Processing product: Rustic Vase Set
Uploading image: https://cdn.sanity.io/images/ri847jqu/production/2219cafc285ec13a2ed3f88aa36cbea852a11735-305x375.png
Image uploaded successfully: image-2219cafc285ec13a2ed3f88aa36cbea852a11735-305x375-png
Category created successfully {
  _createdAt: '2025-02-01T15:09:22Z',
  _id: 'plant-pots-3',
  _rev: 'vHv8Ju1iKPqBAbonebJDd0',
  _type: 'category',
  _updatedAt: '2025-02-01T15:09:22Z',
```



✅ Imported product: Rustic Vase Set

Processing product: Bed

Uploading image: <https://cdn.sanity.io/images/ri847jqu/production/b6551d6252d30e222875fbcd5d91c6edca324a1d-3518x3238.jpg>

Image uploaded successfully: image-b6551d6252d30e222875fbcd5d91c6edca324a1d-3518x3238-jpg

✅ Imported product: Bed

Processing product: Wood Chair

Uploading image: <https://cdn.sanity.io/images/ri847jqu/production/2c7c233904e6c98e27a2156dce5f008b1554d25d-2831x3534.jpg>

Image uploaded successfully: image-638c3039e0746f217acbdc325a2847939d92b08-2831x3534-jpg

✅ Imported product: Wood Chair

Processing product: Retro Vibe

Uploading image: <https://cdn.sanity.io/images/ri847jqu/production/6e986db673401a7d253e8bdee86aceb4b71b4895-4144x4144.jpg>

Image uploaded successfully: image-326af1f82195f9f3a6a9e8c882691aa81f2d0617-4144x4144-jpg

✅ Imported product: Retro Vibe

Processing product: The Lucky Lamp

Uploading image: <https://cdn.sanity.io/images/ri847jqu/production/c10750f3c15f8807251977fe5e042b0a6cee55ab-305x375.png>

Image uploaded successfully: image-c10750f3c15f8807251977fe5e042b0a6cee55ab-305x375-png

```
Category created successfully {
  _createdAt: '2025-02-01T15:10:46Z',
  _id: 'ceramics-7',
  _rev: 'vHv8Ju1iKPqBAbonebJcWw',
  _type: 'category',
  _updatedAt: '2025-02-01T15:10:46Z',
  name: 'Ceramics',
  slug: { _type: 'slug', current: 'ceramics' }
}
```

✅ Imported product: The Lucky Lamp

Processing product: Pure Aura

Uploading image: <https://cdn.sanity.io/images/ri847jqu/production/db6d068faf67dab6145cadcd8e572944c6b9e935-3648x5472.jpg>

Image uploaded successfully: image-db6d068faf67dab6145cadcd8e572944c6b9e935-3648x5472-jpg

✅ Imported product: Pure Aura

Processing product: Zen Table

Uploading image: <https://cdn.sanity.io/images/ri847jqu/production/73e36a9afc074b751d9f2693d2b251ba12b36276-3433x3064.jpg>

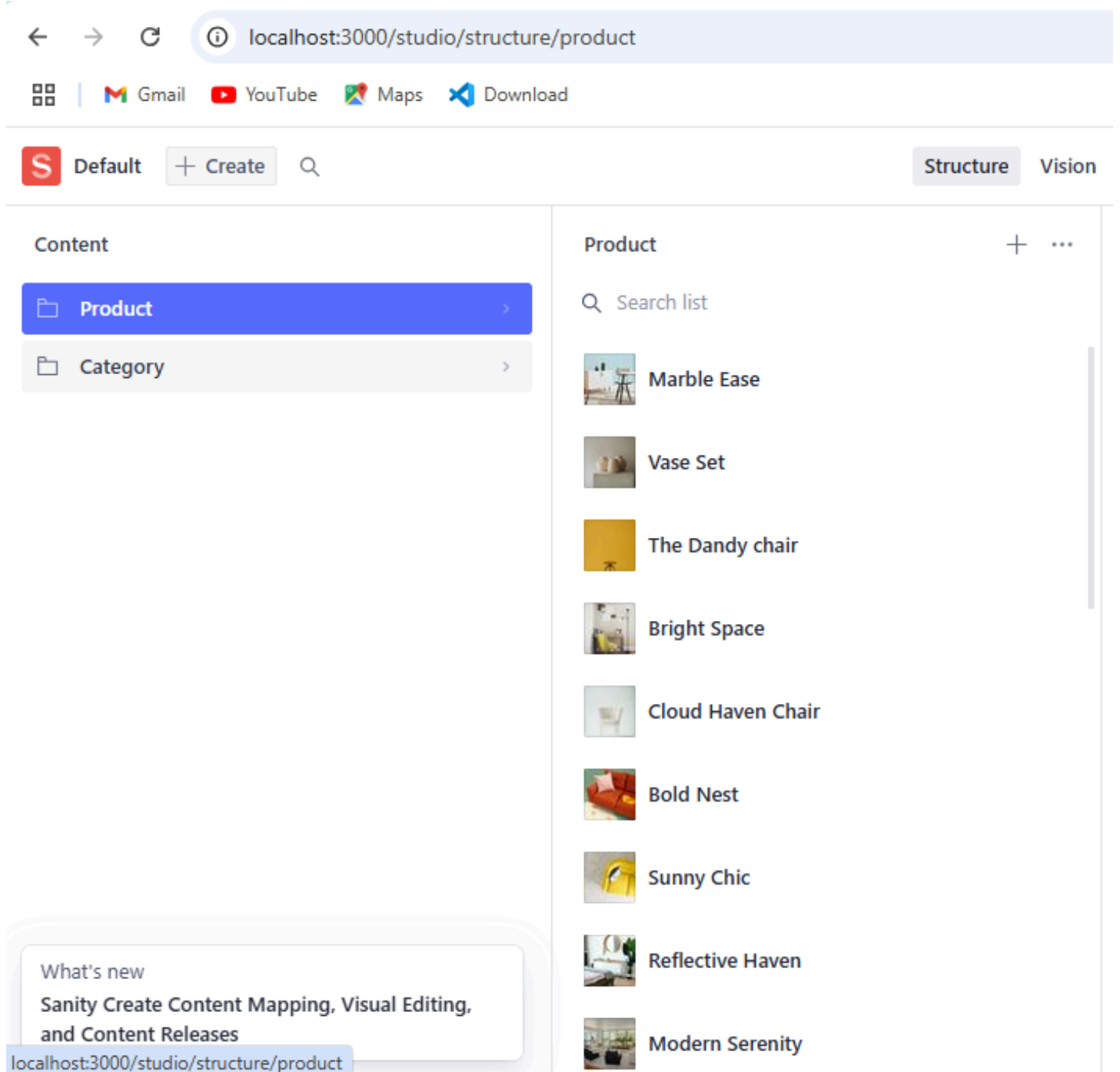
Image uploaded successfully: image-2e9468c5de6d27c850d6cba74ee16bcbf200d77a-3433x3064-jpg

```
Category created successfully {
  _createdAt: '2025-02-01T15:11:35Z',
  _id: 'tables-9',
  _rev: 'vHv8Ju1iKPqBAbonebJggC',
}
```

```
}
✅ Imported product: Zen Table
Processing product: Tropical Vibe
Uploading image: https://cdn.sanity.io/images/ri847jqu/production/9dd7acc17365cd290757246a27b64a3d817cbe13-5501x3095.jpg
Image uploaded successfully: image-f575d30bda78fa7680a74e131a9b25add2247210-5501x3095-jpg
✅ Imported product: Tropical Vibe
Processing product: Sleek Living
Uploading image: https://cdn.sanity.io/images/ri847jqu/production/b7565060672485ede215bbbb4d6b063b1a83b49-3024x4032.jpg
Image uploaded successfully: image-b7565060672485ede215bbbb4d6b063b1a83b49-3024x4032-jpg
✅ Imported product: Sleek Living
Processing product: Serene Seat
Uploading image: https://cdn.sanity.io/images/ri847jqu/production/19920122d503e8a40aa994d43f5f090547b48e9d-3777x5990.jpg
Image uploaded successfully: image-19920122d503e8a40aa994d43f5f090547b48e9d-3777x5990-jpg
Category created successfully {
  _createdAt: '2025-02-01T15:13:48Z',
  _id: 'cutlery-12',
  _rev: 'EIzlwLlFJokSDqejrV8B',
  _type: 'category',
  _updatedAt: '2025-02-01T15:13:48Z',
  name: 'Cutlery',
  slug: { _type: 'slug', current: 'cutlery' }
}
✅ Imported product: Serene Seat
Processing product: Nordic Elegance
Uploading image: https://cdn.sanity.io/images/ri847jqu/production/c371d0a7592905e368cf906ccc918d41033e7a85-4140x5520.jpg
Image uploaded successfully: image-e1cf4b8cb7c99ed0f181d33a70296dfe2819cf46-4140x5520-jpg
✅ Imported product: Nordic Elegance
Processing product: TimberCraft
Uploading image: https://cdn.sanity.io/images/ri847jqu/production/35a8072e40a8b80142ff8e96e0f3af21811f5b9f-3264x4928.jpg
Image uploaded successfully: image-0ede9a4310fc373fcadf2707a5d4f145cf7a8806-3264x4928-jpg
✅ Imported product: TimberCraft
Processing product: Timeless Elegance
Uploading image: https://cdn.sanity.io/images/ri847jqu/production/09ddef74e0be13da2648eceed6504e317ce47847-6000x4000.jpg
Image uploaded successfully: image-3571cf713dde874e2c96085f8538af6bf8af8845-6000x4000-jpg
✅ Imported product: Timeless Elegance
Processing product: Modern Serenity
Uploading image: https://cdn.sanity.io/images/ri847jqu/production/9074eb33bece6914927f6e78eb520d7b0695f77c-3715x2362.jpg
```

```
Image uploaded successfully: image-29e3993cfc3f28149ef0615a5cf933a8100e243b-4000x3907-jpg
✅ Imported product: Sunny Chic
Processing product: Bold Nest
Uploading image: https://cdn.sanity.io/images/ri847jqu/production/75ef174ddda5862f2b2e869b169d8c0fbc7ce15b-4160x4160.jpg
Image uploaded successfully: image-4d749d58f7cf906766229e32feb894a85ac88756-4160x4160-jpg
✅ Imported product: Bold Nest
Processing product: Cloud Haven Chair
Uploading image: https://cdn.sanity.io/images/ri847jqu/production/78683aab6c58d367f91e411fbe2aea5f06a9a9b7-3264x4902.jpg
Image uploaded successfully: image-a309259aad5aa63a3b438ed9c42339a1c8375a3a-3264x4902-jpg
✅ Imported product: Cloud Haven Chair
Processing product: Bright Space
Uploading image: https://cdn.sanity.io/images/ri847jqu/production/d32066d0882c1632e027b053564115ed460e09eb-3393x5090.jpg
Image uploaded successfully: image-d32066d0882c1632e027b053564115ed460e09eb-3393x5090-jpg
✅ Imported product: Bright Space
Processing product: The Dandy chair
Uploading image: https://cdn.sanity.io/images/ri847jqu/production/e2cc69a6c9f9d1ff1fcfbe839844240e16f5f758-305x375.png
Image uploaded successfully: image-e2cc69a6c9f9d1ff1fcfbe839844240e16f5f758-305x375-png
✅ Imported product: The Dandy chair
Processing product: Vase Set
Uploading image: https://cdn.sanity.io/images/ri847jqu/production/8795ff7bdb3d810dbe0c3eb52cd22116c9115bce-305x375.png
Image uploaded successfully: image-8795ff7bdb3d810dbe0c3eb52cd22116c9115bce-305x375-png
✅ Imported product: Vase Set
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: '2025-02-01T15:27:50Z',
  _id: 'crockory-24',
  _rev: 'vHv8Ju1iKPqBAbonebL51Q',
  _type: 'category',
  _updatedAt: '2025-02-01T15:27:50Z',
  name: 'Crockory',
  slug: { _type: 'slug', current: 'crockory' }
}
✅ Imported product: Marble Ease
✅ Data import completed!
```

□ Data successfully displayed in the studio



□ Populated Sanity CMS fields.

←

→

↺

localhost:3000/studio/structure/product:product-24

☆

☰

👤

🗖

Gmail

YouTube

Maps

Download

All Bookmarks

S

Default

+ Create

🔍

Structure

Vision

Schedules

⚡

🕒

👤

🕒

Content

Product

Category

Product

Search list

Marble Ease

Vase Set

The Dandy chair

Bright Space

Cloud Haven Chair

Bold Nest

Sunny Chic

Reflective Haven

Modern Serenity

Marble Ease

Product

Marble Ease

Category

Crockory

Title

Marble Ease

Slug

Published 28 min. ago

Publish

What's new

Sanity Create Content Mapping, Visual Editing, and Content Releases

←

→

↺

localhost:3000/studio/structure/category:crockory-24

☆

☰

👤

🗖

Gmail

YouTube

Maps

Download

All Bookmarks

S

Default

+ Create

🔍

Structure

Vision

Schedules

⚡

🕒

👤

🕒

Content

Product

Category

Category

Search list

Crockory

Cutlery

Tables

Ceramics

Plant Pots

Chairs

Tableware

Crockory

Category

Crockory

Name

Crockory

Slug

crockory

Generate

Published 29 min. ago

Publish

What's new

Sanity Create Content Mapping, Visual Editing, and Content Releases

localhost:3000/studio/vision

Default + Create

Structure Vision Schedules

DATASET: production API VERSION: Other CUSTOM API VERSION: v2025-02-01 PERSPECTIVE: raw QUERY URL [COPY TO CLIPBOARD]: https://uz5dxb.api.sanity.io/v2025-02-01/data/query/pr

QUERY

```
1 *[_type == "product"]
```

PARAMS

```
1 {
2
3 }
```

RESULT

```
[...] 24 items
  0: {...} 15 properties
    slug: {...} 2 properties
      current: the-poplar-suede-sofa
      _type: slug
    dimensions: {...} 3 properties
      depth: 50cm
      width: 110cm
      height: 110cm
    tags: [...] 1 item
      0: popular products
      price: 980
      _createdAt: 2025-02-01T15:09:05Z
      quantity: 50
    features: [...] 3 items
      0: Premium material
```

Fetch Listen

Execution: 5ms End-to-end: 770ms

Save result as JSON CSV

## Fetching Data from Sanity

```
const Ceramics = () => {
  const [products, setProducts] = useState<Product[]>([]);
  const [visibleCount, setVisibleCount] = useState(4); // Controls how many products are visible
  const productsPerPage = 4; // Number of products to load each time

  useEffect(() => {
    const fetchProducts = async () => {
      const query = `*[_type == "product" && category->name in ["Ceramics","Plant Pots"]]{
        _id,
        name,
        price,
        "imageUrl": image.asset->url
      }`;

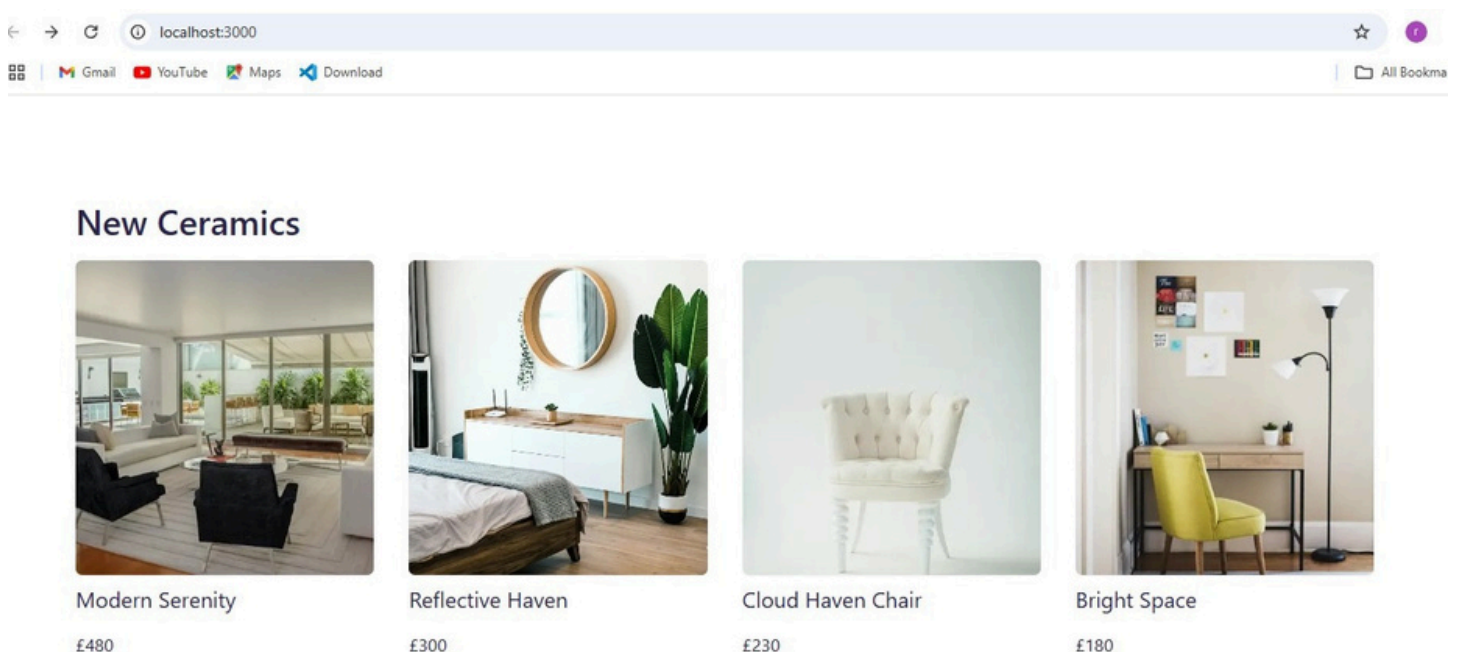
      try {
        const result: Product[] = await client.fetch(query);
        setProducts(result);
      } catch (error) {
        console.error("Error fetching products:", error);
      }
    };

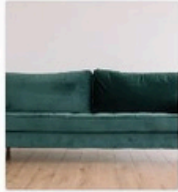
    fetchProducts();
  }, []);
};
```



```
const Ceremics = () => {
  return (
    <div id="ceramics" className="w-full md:px-20 p-10 text-[#2A254B]">
      <div className="text-2xl font-semibold sm:text-[32px] my-5 text-[#2A254B]">
        New Ceramics
      </div>
      <div className="text-[#2A254B] grid grid-cols-1 sm:grid-cols-2 lg:grid-cols-4 gap-8">
        {products.slice(0, visibleCount).map((prod) => (
          <div className="space-y-4" key={prod._id}>
            <Link href={` /product/${prod._id}`}>
              <Image
                src={prod.imageUrl}
                alt={prod.name}
                height={375}
                width={305}
                className="object-cover h-72 rounded-md shadow-sm hover:shadow-xl transition-all mb-2"
              />
              <p className="text-xl">{prod.name}</p>
            </Link>
            <p className="text-base">£{prod.price}</p>
          </div>
        ))}
      </div>
      <div className="flex justify-center items-center mt-16">
        {products.length > productsPerPage && (
          <Button
            variant={"mybutton"}
            className="w-[170px] h-[56px]"
            onClick={handleViewToggle}
          />
        )}
      </div>
    </div>
  )
}
```

□ Data successfully displayed in the Frontend:





The Poplar suede sofa  
\$980



Tropical Vibe  
\$550



Sleek Living  
\$300



Serene Seat  
\$350



Nordic Elegance  
\$280



TimberCraft  
\$320



Timeless Elegance  
\$320



Modern Serenity  
\$480



Reflective Haven  
\$300



Sunny Chic  
\$400



Bold Nest  
\$260



The Steel Chair  
\$250