

DAY-3 API INTEGRATION AND DATA MIGRATION

Integrating Sanity API with Next.js:

Step-1: Installing Next.js npx [create-next-app@14.2.15](#)

Step 2: Installing Sanity: npm install @sanity/client

Step 3: Setting Up Environment Variables:

I created a .env file in the root of my project to store my environment variables securely. Inside this file, I defined the following variables:

NEXT_PUBLIC_SANITY_PROJECT_ID=your-project-id NEXT_PUBLIC_SANITY_DATASET=your-dataset-name
SANITY_API_TOKEN=your-api-token

I used these environment variables to configure my connection with the Sanity API.

```
.env.local × JS importData.mjs TS products.ts
.env.local
1 NEXT_PUBLIC_SANITY_PROJECT_ID="cijmx044"
2 NEXT_PUBLIC_SANITY_DATASET="production"
3 SANITY_API_TOKEN="sk0RTxKjqIoqMzpQmP3sVT7AS5EzIT2axt3cDPn7Gj36BH7
4 RgkG6QSYUZ8NV2rv3z1aIEu0kKZMGw6WBd5v98o6I6V0Tnn008QwL6bzKMH80eQy
5 MkG1lC1ysp9Ph2h6gtg93DjLx1HrWs5A1akY3jcccfzCpDk0qv5a9vP4b7GRDbuejpy0"
6
```

Step 4: Generating API Token

The screenshot shows the Sanity Studio interface for the 'gen-ecom' project. The 'API' tab is selected in the top navigation bar. On the left sidebar, 'Webhooks', 'CORS origins', and 'Tokens' are listed. The main content area shows a table of API tokens. A new token has been generated, and a yellow box highlights the token string, which is: `sk0RTxKjqIoqMzpQmP3sVT7AS5EzIT2axt3cDPn7Gj36BH7RgkG6QSYUZ8NV2rv3z1aIEu0kKZMGw6WBd5v98o6I6V0Tnn008QwL6bzKMH80eQyMkG1lC1ysp9Ph2h6gtg93DjLx1HrWs5A1akY3jcccfzCpDk0qv5a9vP4b7GRDbuejpy0`. A message above the token says: 'Copy the token below - this is your only chance to do so!'.

NAME	PERMISSIONS	CREATED
Gen-Ecom	Developer	0 seconds

Step-5 :Project ID

S

Muhammad Tariq Mahboob

gen-ecom

30 days left in trial

GE

PLAN

Growth Trial

STATUS

Active

PROJECT ID

cjms044

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

(2 in)

Get me

Step 5: Defining Schema Types

```
sanity > schemaTypes > TS products.ts > product
1  import { defineType } from "sanity"
2
3  export const product = defineType({
4    name: "product",
5    title: "Product",
6    type: "document",
7    fields: [
8      {
9        name: "title",
10       title: "Title",
11       validation: (rule) => rule.required(),
12       type: "string"
13     },
14     {
15       name: "description",
16       type: "text",
17       validation: (rule) => rule.required(),
18       title: "Description",
19     },
20     {
21       name: "productImage",
22       type: "image",
23       validation: (rule) => rule.required(),
24       title: "Product Image"
25     }
26   ]
27 })
```

Step-6: ImportData.mjs

```
scripts > .\ importData.mjs > importProducts > response
8  const __filename = fileURLToPath(import.meta.url);
9  const __dirname = path.dirname(__filename);
10 dotenv.config({ path: path.resolve(__dirname, '../.env.local') });
11
12 // Create Sanity client
13 const client = createClient({
14   projectId: "cijmx044",
15   dataset: "production",
16   useCdn: true,
17   apiVersion: '2025-01-13',
18   token: "sk0RTxKjqIoqMzP3sVT7AS5EzIT2axt3cDPPr76j36BH7RgkG6QSYUZ8NV2rv3z1aIEu0kKZMGw6WBd5v",
19 });
20
21 async function uploadImageToSanity(imageUrl) {
22   try {
23     console.log(`Uploading image: ${imageUrl}`);
```

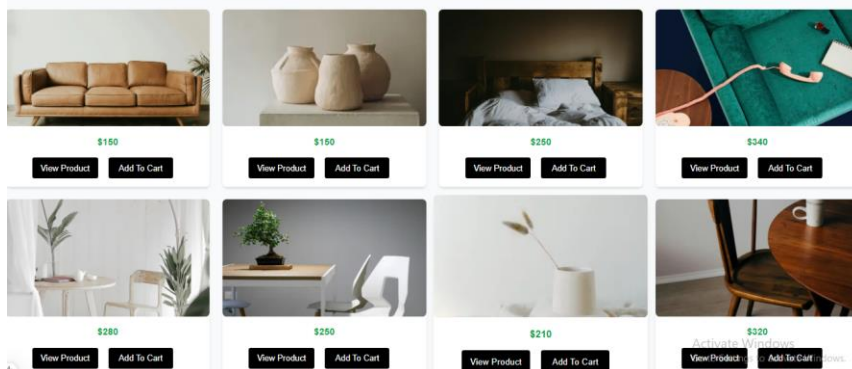
Step-7: Data Fetch

```
{ } package.json > type
1  {
2    "name": "gen-ecom",
3    "version": "0.1.0",
4    "private": true,
5    "type": "module",
6    "scripts": {
7      "dev": "vite",
8      "build": "vite build",
9      "preview": "vite preview",
10     "import": "node importData.mjs",
11     "importProducts": "node importProducts.mjs",
12     "response": "node response.mjs"
13   }
14 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS


```
ely and sophisticated living experience. Make every day feel like a sunlit escape.',
  image: {
    _type: 'image',
    asset: {
      _ref: 'image-ed5874fb4498ee4ff897075bad504056cb87388c-4000x3907-jpg'
    }
  },
  isNew: false,
  price: 400
  price: 400
  price: 400
}
Data import completed successfully!
E:\gen-ecom>
```

View Product || Add to cart



Your Cart

Your Cart



Price: \$350

Quantity: 7

-

7

+

Remove

Subtotal

\$2450.00

Discount (20%)

-\$490.00

Delivery Fee

\$15.00

Total

\$1975.00

Go to Checkout