# DAY 3: API INTEGRATION AND DATA MIGRATION

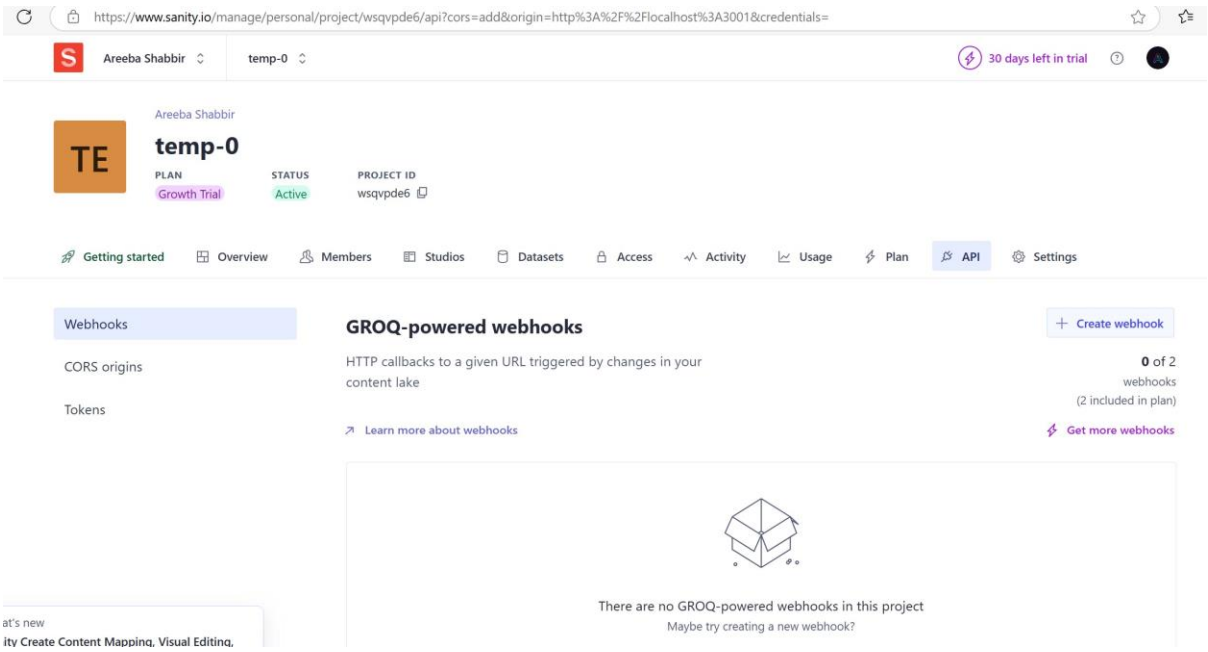
- **Sanity Account Setup**:
  - Sign up for a Sanity account at Sanity.io.
  - Create a new project in the Sanity dashboard.

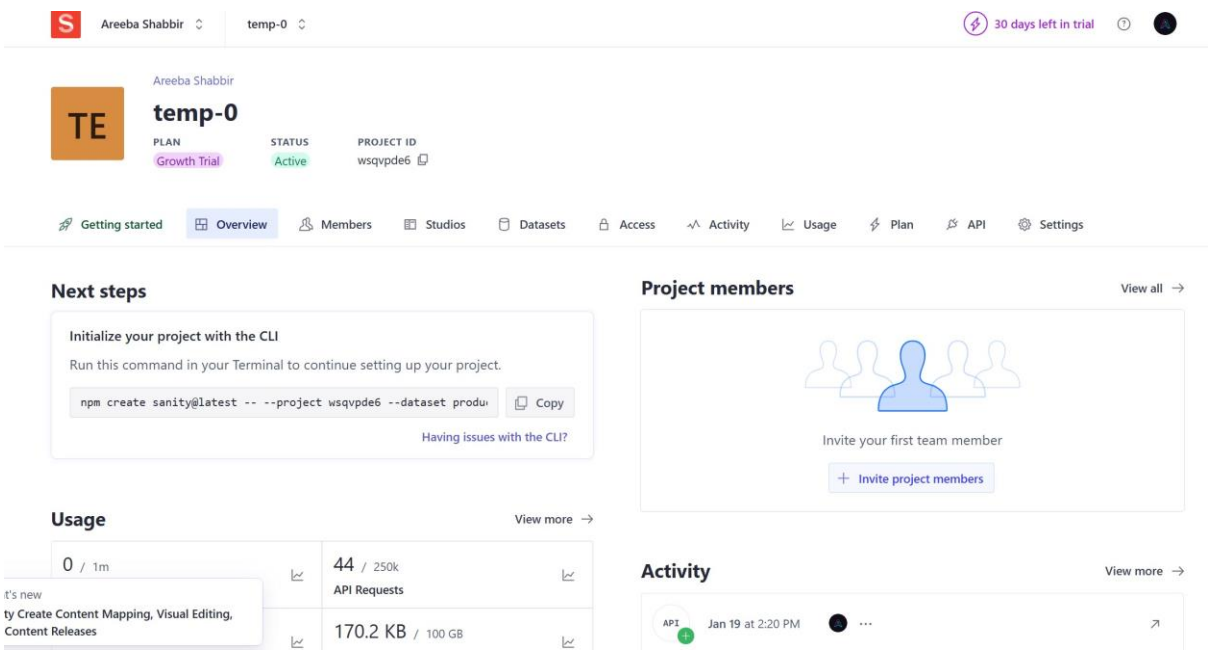
- **API Token Generation**:
  - Go to the *API* section in your project settings and generate a new API token for secure access.

- **Sanity Client Setup**:
  - Install the Sanity client in your project and configure it using the project ID and API token.

- **Data Fetching**:
  - Use the Sanity client to fetch product, order, or user data for your project.

- **Security**:
  - Keep your API token secure and avoid exposing it on the frontend. Use environment variables for sensitive data.






## Sanity Integration in My Project

1. **Setting Up Sanity**:
   - I created a Sanity account and generated an API token to interact with my Sanity CMS.
   - The Sanity client was installed and configured in my project using the project ID and API token.
   -
2. **Using data-migration.mjs**:
   - The data-migration.mjs file was created to automate the migration of product data into the Sanity CMS.
   - It is used for importing product details (e.g., product names, images, and prices) into Sanity, streamlining the process of adding or updating multiple products at once.
   -
3. **Configuring product.ts**:
   - In product.ts, I defined the structure of my product data, including important fields like name, description, price, and images.
   - The file also handles fetching product data from Sanity using queries and ensures proper data typing through TypeScript.
   -
4. **Data Migration Process**:
   - I ran the data-migration.mjs script to migrate product data into Sanity, allowing for easy management and updates of product information within the CMS.

EXPLORER

MY-FIGMA-PROJECT
- .next
- node_modules
- public
- scripts
  - data-migration.mjs U
- src
  - app
  - sanity
    - lib
    - schemaTypes
      - index.ts U
      - product.ts U
    - env.ts U
    - structure.ts U
- .env.local
- .gitignore
- eslint.config.mjs
- next-env.d.ts
- next.config.ts M
- package-lock.json M
- package.json M
- postcss.config.mjs
- README.md
- sanity.cli.ts U
- sanity.config.ts U
- tailwind.config.ts
- tsconfig.json

OUTLINE
TIMELINE
NPM SCRIPTS

TS product.ts U | JS data-migration.mjs U | TS index.ts U

scripts > data-migration.mjs > importData > response

```
37  async function importData() {
38    try {
39      console.log('Migrating data, please wait...');
40
41      // Fetch products from the API
42      const response = await axios.get('https://template-0-beta.vercel.app/api/product');
43      const products = response.data;
44
45      console.log('Products fetched:', products);
46
47      for (const product of products) {
48        let imageRef = null;
49
50        if (product.imagePath) {
51          imageRef = await uploadImageToSanity(product.imagePath);
52        }
53
54        const sanityProduct = {
55          _type: 'product',
56          id: product.id,
57          name: product.name,
58          category: product.category,
59          description: product.description,
60          discountPercentage: product.discountPercentage,
61          isFeaturedProduct: product.isFeaturedProduct,
62          stockLevel: product.stockLevel,
63          price: parseFloat(product.price),
64          image: imageRef
65          ? {
66            _type: 'image',
67            asset: {
68              _type: 'reference',
69              _ref: imageRef,
70            },
71          }
```

MY-FIGMA-PROJECT
- .next
- node_modules
- public
- scripts
  - data-migration.mjs U
- src
  - app
  - sanity
    - lib
    - schemaTypes
      - index.ts U
      - product.ts U
    - env.ts U
    - structure.ts U
- .env.local
- .gitignore
- eslint.config.mjs
- next-env.d.ts
- next.config.ts M
- package-lock.json M
- package.json M
- postcss.config.mjs
- README.md
- sanity.cli.ts U
- sanity.config.ts U
- tailwind.config.ts
- tsconfig.json

OUTLINE
TIMELINE
NPM SCRIPTS

src > sanity > schemaTypes > product.ts > default

```
1  export default {
2    name: 'product',
3    title: 'Product',
4    type: 'document',
5    fields: [
6      {
7      name: 'id',
8      title: 'ID',
9      type: 'string',
10     },
11     {
12     name: 'name',
13     title: 'Name',
14     type: 'string',
15     },
16     {
17     name: 'image',
18     title: 'Image',
19     type: 'image',
20     },
21     {
22     name: 'imagePath',
23     title: 'Image Path',
24     type: 'url',
25     },
26     {
27     name: 'price',
28     title: 'Price',
29     type: 'number',
30     },
31     {
32     name: 'description',
33     title: 'Description',
34     type: 'text',
35     },
36     {
```

## Uploading Images and Data to Sanity

1. **Uploading Data**:
    - I successfully uploaded product data, including product names, descriptions, and pricing, to Sanity using the CMS interface and the `data-migration.mjs` script.
    - The data is structured and stored within Sanity under appropriate document types (e.g., product details).
    -
2. **Uploading Images**:
    - I uploaded product images to Sanity's asset library, ensuring that they are linked to the respective products.
    - The image URLs are now accessible and used in the product listings on the frontend.
    -
3. **Data and Image Management**:
    - All product data, including images, is now easily manageable and updateable through the Sanity dashboard, allowing for efficient content management.

EXPLORER

MY-FIGMA-PROJECT
- node_modules
- notes
- public
- scripts
  - data-migration... U
- src
  - app
  - sanity
    - lib
    - schemaTypes
      - index.ts U
      - product.ts U
    - env.ts U
    - structure.ts U
- .env.local
- .gitignore
- eslint.config.mjs
- next-env.d.ts
- next.config.ts M
- package-lock.json M
- package.json M
- postcss.config.mjs
- README.md
- sanity.cli.ts U
- sanity.config.ts U
- tailwind.config.ts
- tsconfig.json

OUTLINE
TIMELINE
NPM SCRIPTS

TS product.ts U | JS data-migration.mjs U | TS index.ts U

scripts > JS data-migration.mjs > importData > response

```
37  async function importData() {
38    try {
39      console.log('Migrating data, please wait...');
40
41      // Fetch products from the API
42      const response = await axios.get('https://template-0-beta.vercel.app/api/product');
43      const products = response.data;
44
```

PROBLEMS | OUTPUT | DEBUG CONSOLE | TERMINAL | PORTS

```
Uploading image: https://images.unsplash.com/photo-1467043153537-a4fba2cd39ef?q=80&w=1919&auto=format&fit=crop&ixlib=rb-4.0.3&ixid=M3wxMjA3fDB8MHxwaG9
0by1wYWdlfHx8fGVufDB8fHx8fA%3D%3D
Image uploaded successfully: image-c4cada82ea92c0feab52e37e54ece957dcefc5ce-1919x2660-jpg
Product created in Sanity: 15
Uploading image: https://images.unsplash.com/photo-1522771739844-6a9f6d5f14af?q=80&w=2071&auto=format&fit=crop&ixlib=rb-4.0.3&ixid=M3wxMjA3fDB8MHxwaG9
0by1wYWdlfHx8fGVufDB8fHx8fA%3D%3D
Image uploaded successfully: image-433a176232584b477a55285eb9ff6bf9d2de4b5c-2071x1381-jpg
Product created in Sanity: 16
Uploading image: https://plus.unsplash.com/premium_photo-1683120852623-143817d6400b?q=80&w=1776&auto=format&fit=crop&ixlib=rb-4.0.3&ixid=M3wxMjA3fDB8M
HxwaG90by1wYWdlfHx8fGVufDB8fHx8fA%3D%3D
Image uploaded successfully: image-ccb5458804bfb0eb4d3e074804e8ef889e96c024-1776x1176-jpg
Product created in Sanity: 17
Uploading image: https://plus.unsplash.com/premium_photo-1661595077028-9ff236368cb5?q=80&w=2069&auto=format&fit=crop&ixlib=rb-4.0.3&ixid=M3wxMjA3fDB8M
HxwaG90by1wYWdlfHx8fGVufDB8fHx8fA%3D%3D
Image uploaded successfully: image-1483cf44fe55027a13051dbc14f6d1d78f3f45cb-2069x1381-jpg
Product created in Sanity: 18
Uploading image: https://plus.unsplash.com/premium_photo-1670869816720-de59bfc2be74?q=80&w=1780&auto=format&fit=crop&ixlib=rb-4.0.3&ixid=M3wxMjA3fDB8M
HxwaG90by1wYWdlfHx8fGVufDB8fHx8fA%3D%3D
Image uploaded successfully: image-834fbc1fcacf82f1e6d39f262213468933af1431-1780x1780-jpg
Product created in Sanity: 19
Uploading image: https://plus.unsplash.com/premium_photo-1692130314358-30f911957d7f?q=80&w=1932&auto=format&fit=crop&ixlib=rb-4.0.3&ixid=M3wxMjA3fDB8M
HxwaG90by1wYWdlfHx8fGVufDB8fHx8fA%3D%3D
Image uploaded successfully: image-c02efe31cc7d519664e3b4c4d4608a8b812109e2-1932x1087-jpg
Product created in Sanity: 20
Uploading image: https://plus.unsplash.com/premium_photo-1661886455585-03ddf4347796?q=80&w=2059&auto=format&fit=crop&ixlib=rb-4.0.3&ixid=M3wxMjA3fDB8M
HxwaG90by1wYWdlfHx8fGVufDB8fHx8fA%3D%3D
Image uploaded successfully: image-432c20d55e31514d09985305e657cb342317d57b-2059x1391-jpg
Product created in Sanity: 21
```

main* | 0 0 0 | Ln 42, Col 84 | Spaces: 2 | UTF-8 | CRLF | {} JavaScript

localhost:3000/studio/structure/product

Home | Shop | About | Contact

Default | Create | Structure | Vision | Schedules | Tasks

Content
- Product

Product

Search list

- Luxury Flower Bed
- Blue Bed — Published 6 min. ago
- Red Bed
- White Bed
- Solid Bed
- Matilda Velvet Bed
- Diondre Chair
- Leisure Sofa Chair Set
- Liberty Center
- Sleek Modern Table
- Replica Table