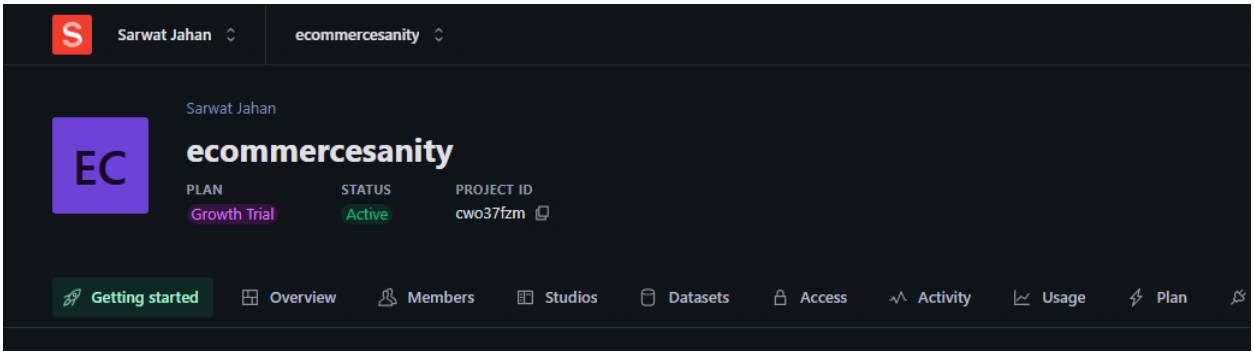
Day3: API Integration and Data Migration for Nike Products Marketplace

This document details the activities undertaken on the third day of the Nike Products Marketplace hackathon.
It includes custom migration, integrating data from Sanity, creating schemas, and presenting data using GROQ queries within a Next.js application.
Each Segment is accompanied by annotated code images with comprehensive explanations of their functions.

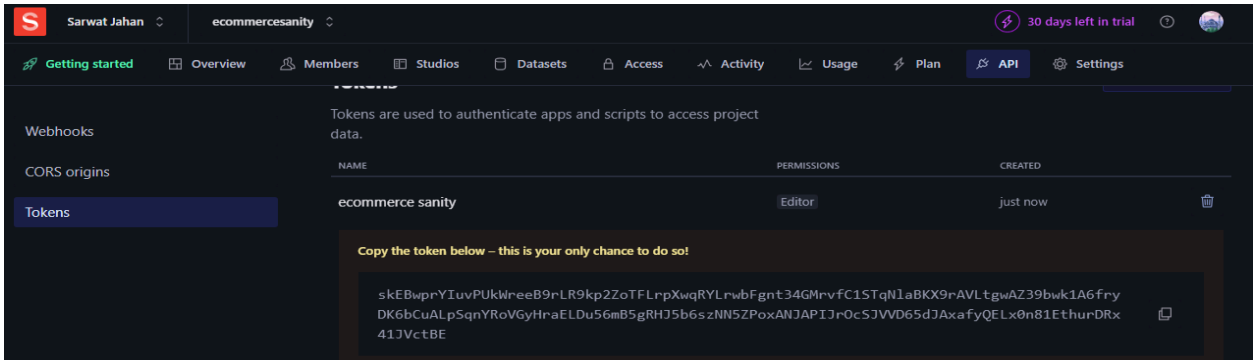
API Integration process:

1.First, created a project in my Sanity account ,named ‘ecommercesanity’.

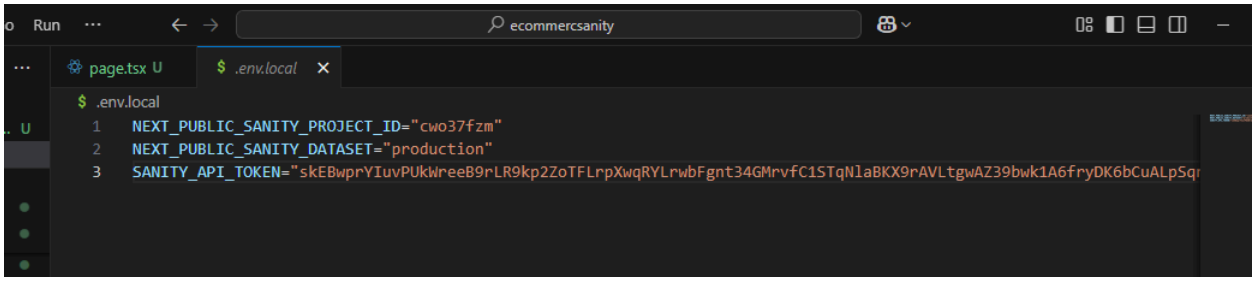
2.Installed Sanity in my Next.js Project folder.



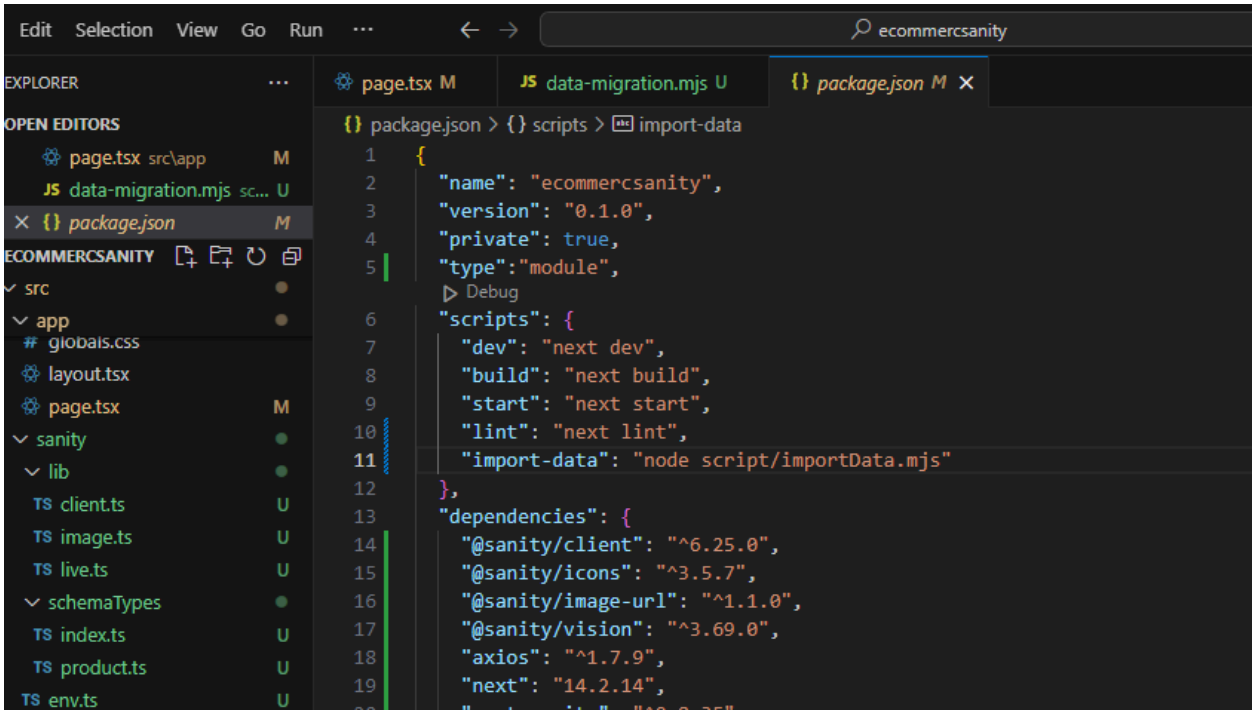
3.Then , I created a Token.



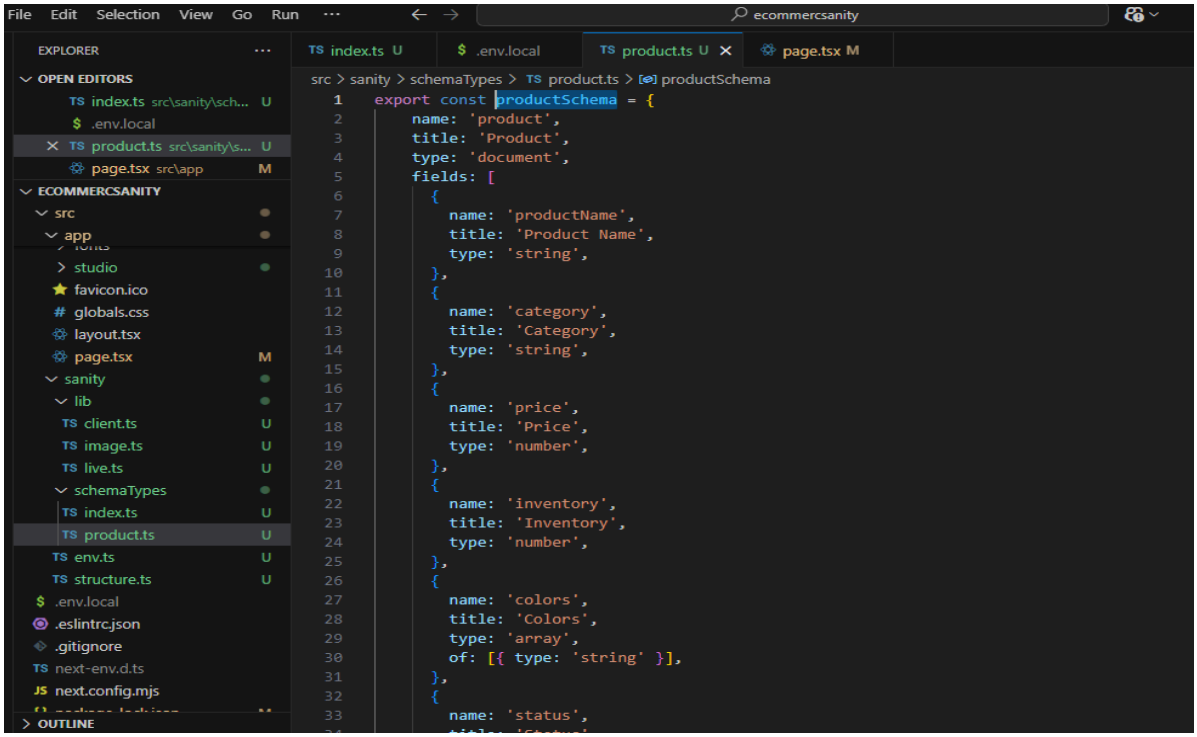
4.Token created in sanity, then pasted in my project.



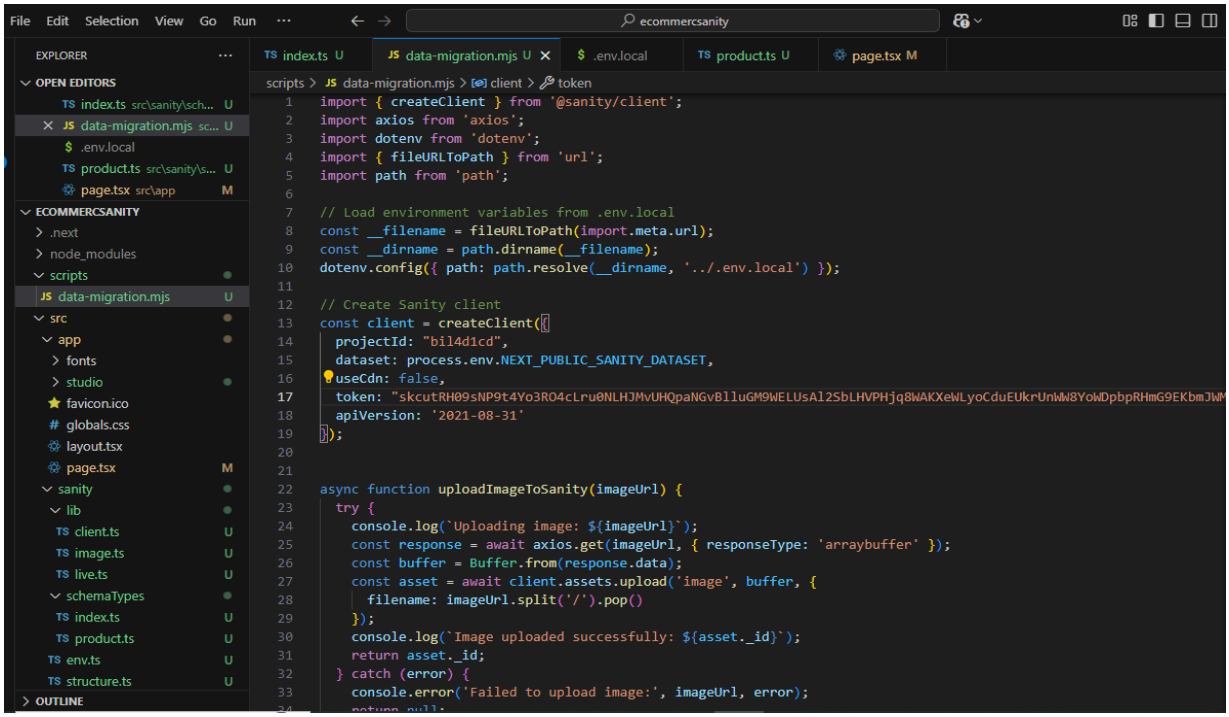
5. Import-Data is added in the package.json file.



6. In SchemaTypes, a file named product.ts is created which contains Product Schema ,provided by the teacher.



7. Created a data-migration.mjs file in scripts folder and add function to integrate API in sanity project.

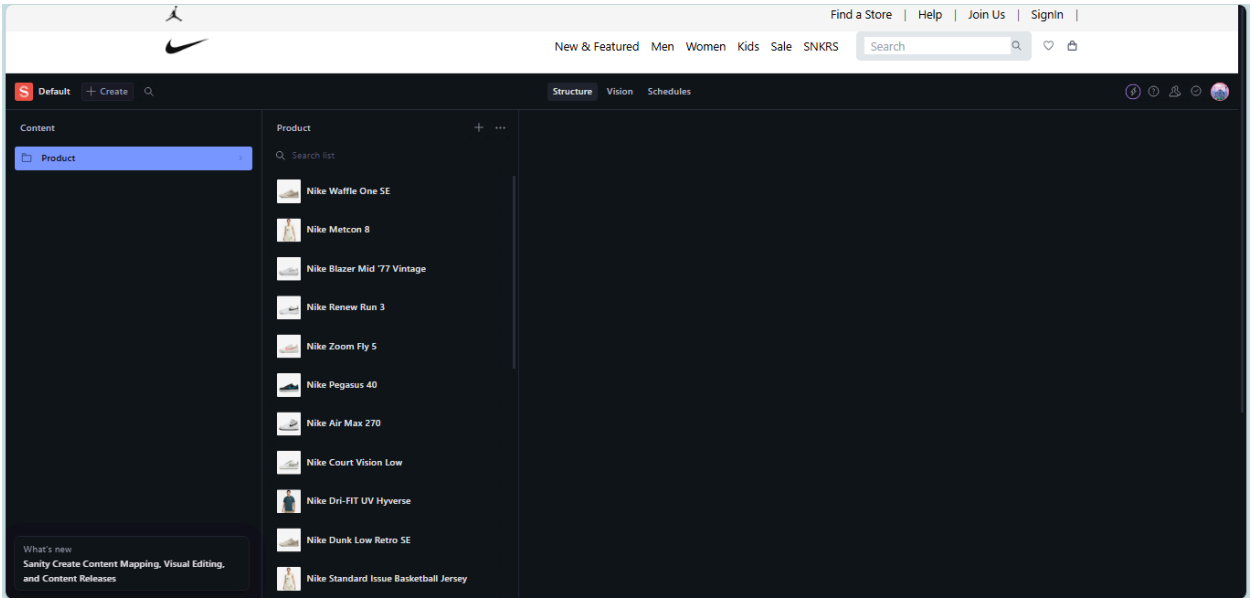


```
import { createClient } from '@sanity/client';
import axios from 'axios';
import dotenv from 'dotenv';
import { fileURLToPath } from 'url';
import path from 'path';

// 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: "bil4d1cd",
  dataset: process.env.NEXT_PUBLIC_SANITY_DATASET,
  useCdn: false,
  token: "skcutRH09sNP9t4Yo3RO4cLru0NLHJMvUHQpaNGvBlluGM9WELUsAl2SbLHVPHjq8WAKXeWLyoCduEUkrUnWW8YoWDpbpRHmG9EKbmJW
  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);
```

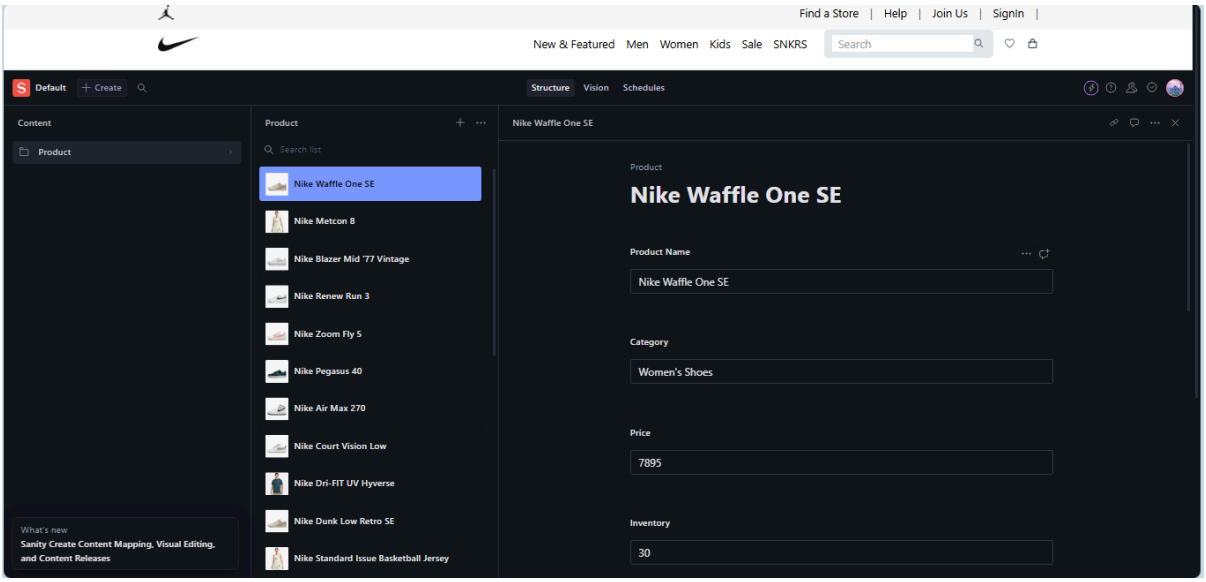
8.After all the important steps for migrating API data in my projects, this command ran in terminal
npm run import -data.

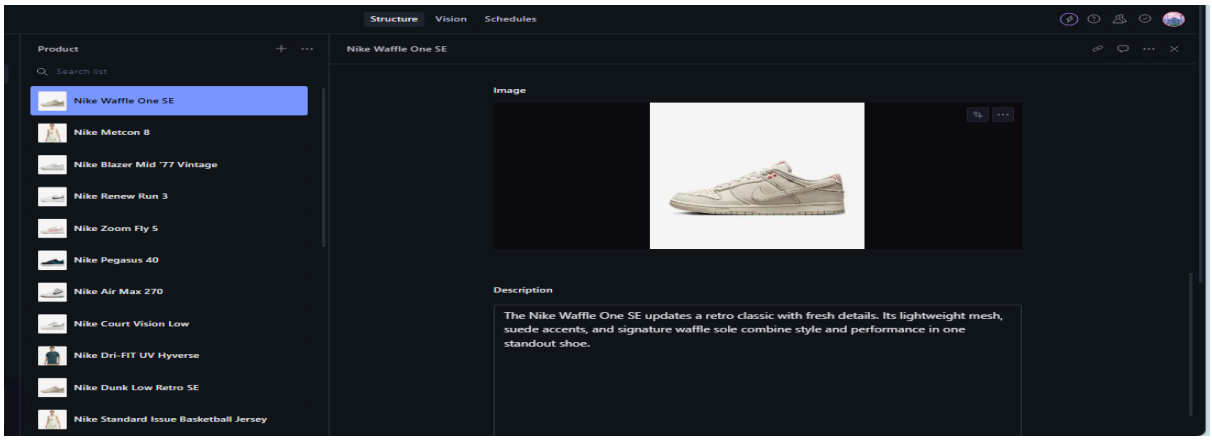
```
> node scripts/importSanityData.mjs

migrating data please wait...
products ==>>  [
  {
    productName: "Nike Air Force 1 Mid '07",
    category: "Men's Shoes",
    price: 10795,
    inventory: 20,
    colors: [ 'White' ],
    status: 'Just In',
    image: 'https://template-03-api.vercel.app/products/1.png',
    description: "The Nike Air Force 1 Mid '07 delivers timeless style with premium leather and mid-cut design. Perfect for everyday wear, it provides
 exceptional comfort and durability. The iconic Air-Sole cushioning adds responsive support for long-lasting performance."
  },
```

9. All the Data through API was integrated in Sanity Studio.



Each Product can see in Studio separately.





Self-Validation Checklist:

Tasks:

API Understanding
Schema Validation
Data Migration
API Integration in Next.js

Submission Preparation:

- Data Fetched and Displayed successfully in Sanity CMS.
- With manually entering data, frontend validation.
- Screenshots are added in file to show my work.

Conclusion:

This report documents the manual data migration process and its integration with Sanity CMS for Nike products, ensuring an optimal user experience for local dropshipping.