Hackathon 3 - Day 3:
API Integration and Data Migration- [Nike store]

Overview:

Day 3 focused on integrating APIs with Sanity CMS, emphasizing the use of GROQ queries and schema updates to support dynamic data needs. Below is a summary of the tasks completed:

Key Objectives:

1. Establish a seamless connection with Sanity CMS to efficiently manage product data.
2. Utilize GROQ queries to dynamically retrieve structured data from Sanity.
3. Revise the Sanity schema to meet the changing requirements of the API.
4. Transfer product data from an external API to Sanity CMS, including uploading associated images.
5. Conduct thorough testing and validation to ensure smooth data integration between the backend and frontend.

1. API Data Migration to Sanity:

Objective:

- Transfer product data, including images, from an external API to Sanity CMS.

Steps Taken:

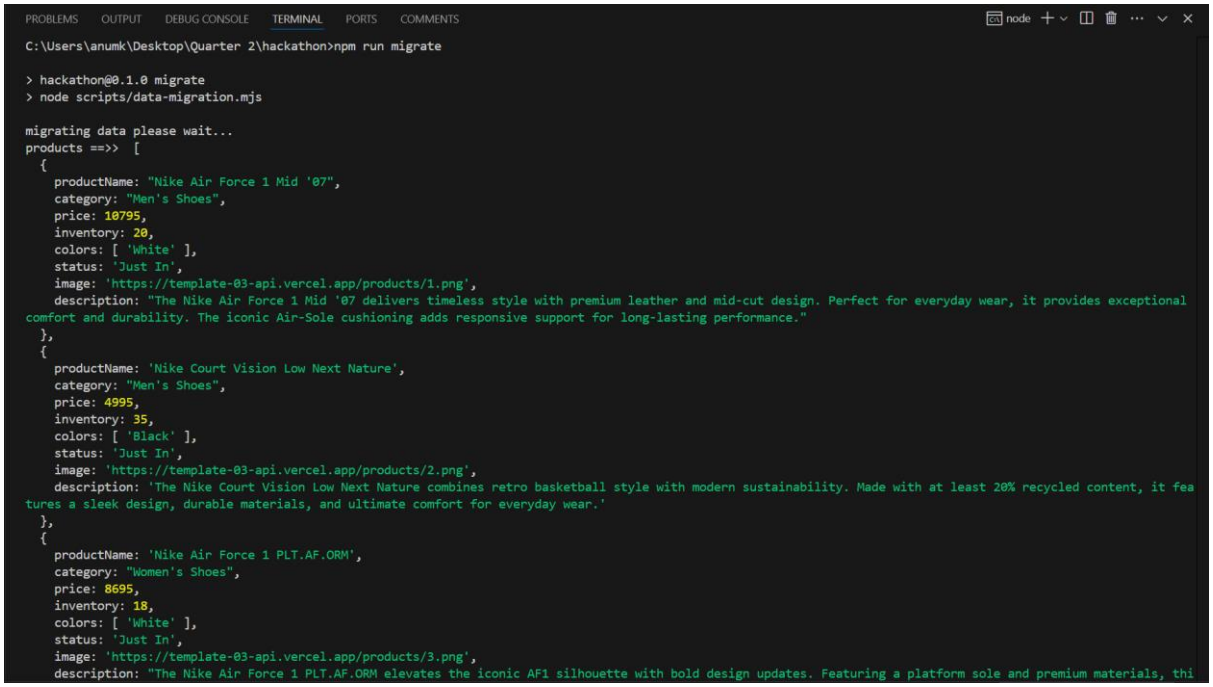
1. **Fetch Product Data:**
    - Utilized Axios to fetch product data from an external API.
    - Example API Endpoint: https://template-03api.vercel.app/api/products.

2. **Upload Images to Sanity:**

    - Used the client.assets.upload method to upload product images to Sanity CMS.
    - Ensured that each image was correctly associated with its respective product.

3. **Create Product Documents in Sanity:**

    - Processed the fetched data by iterating over each product and creating corresponding documents in Sanity CMS using the client.create() method.



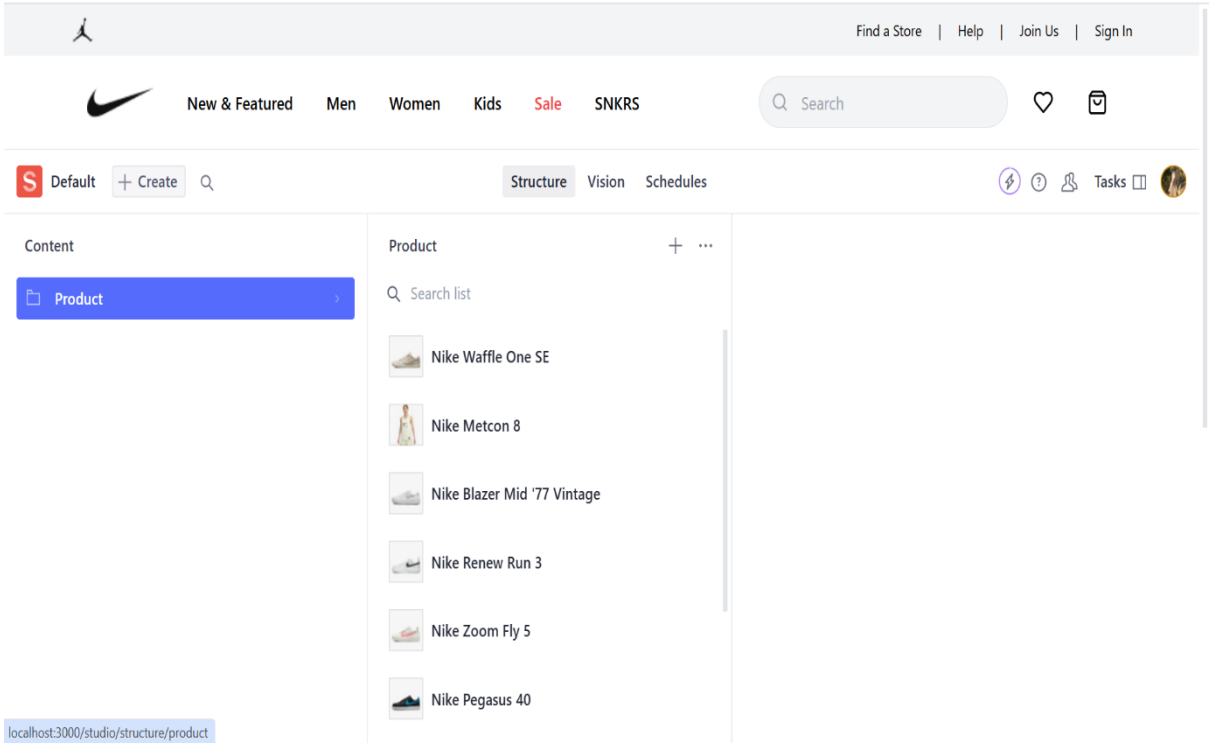
2. Sanity CMS Integration:

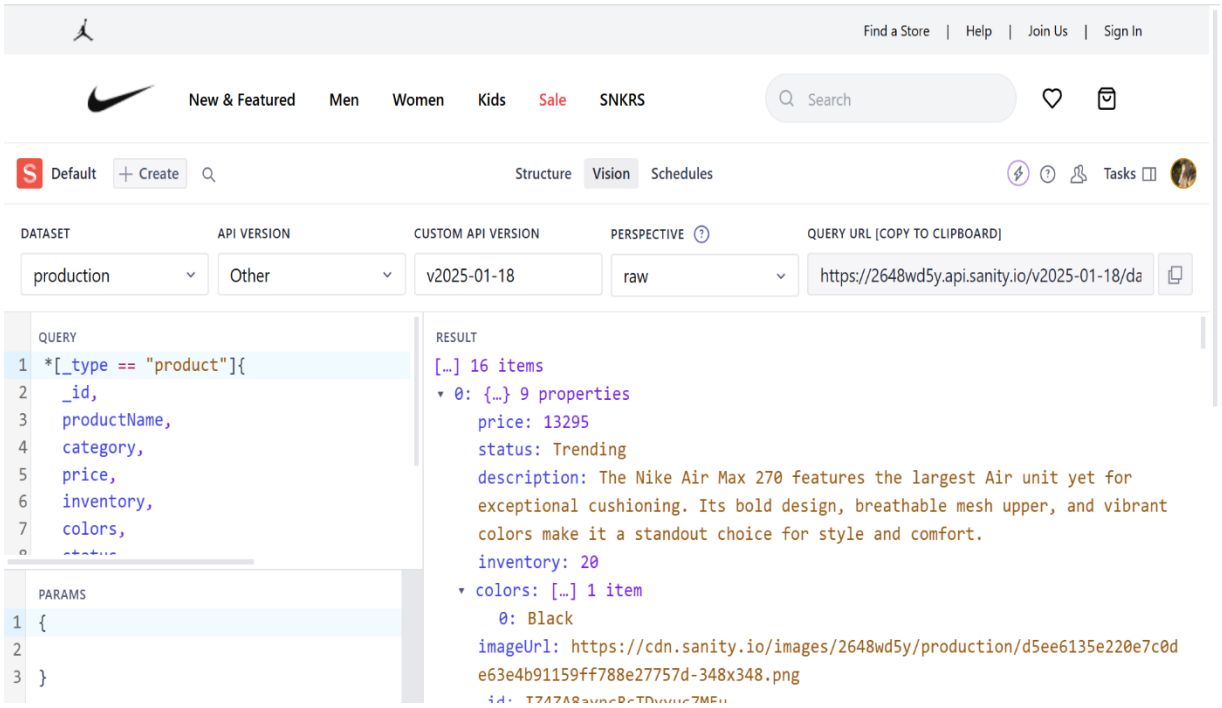
<u>Connection Setup:</u>

- Successfully integrated the @sanity/client library into the project for seamless communication with the Sanity CMS.
- Configured the client with project-specific credentials, including projectId, dataset, and apiVersion, ensuring secure and efficient access.

Purpose:

To streamline product management and take advantage of Sanity's powerful features, including real-time updates, flexible content modeling, and efficient image optimization.





```tsx
import React, { useEffect, useState } from 'react';
import Card from './cards/card';
import { client } from '../sanity/lib/client';

 export interface Product {
  _id: number;
  productName: string;
  category: string;
  price: string;
  colors: string[];
  imageUrl: string;
  description: string;
}


const ProductList = () => {
    const [products, setProducts] = useState<Product[]>([]);
    const [loading, setLoading] = useState<boolean>(true);
    const [error, setError] = useState<string | null>(null);

    useEffect(() => {
      const fetchProducts = async () => {
        const query = `
          *[_type == "product"]{
            _id,
            productName,
            category,
            price,
            inventory,
            colors,
            status,
            "imageUrl": image.asset->url,
            description
          }
        `;

        try {
          setLoading(true);
          const result = await client.fetch(query);
          setProducts(result);
        } catch (err) {
          console.error(err);
          setError('Failed to fetch products');
        } finally {
          setLoading(false);
        }
      };

      fetchProducts();
    }, []);

    if (loading) {
      return <p>Loading...</p>;
    }

    if (error) {
      return <p>{error}</p>;
    }

  return (
    <div className="grid grid-cols-1 md:grid-cols-3 lg:grid-cols-3 pb-10 border-b-2">
      {products.map((product) => (
        <Card
          key={product._id}
          id={product._id}
          category={product.category}
          productName={product.productName}
          description={product.description}
          colors={product.colors}
          price={product.price}
          imageUrl={product.imageUrl}
        />
      ))}
    </div>
  );
};

export default ProductList;

```

Find a Store | Help | Join Us | Sign In

New & Featured Men Women Kids Sale SNKRS

Search

New (500)

Hide Filters Sort By

- Shoes
- Sports Bras
- Tops & T-Shirts
- Hoodies & Sweatshirts
- Jackets
- Trousers & Tights
- Shorts
- Tracksuits
- Jumpsuits & Rompers
- Skirts & Dresses
- Socks
- Accessories & Equipment

Gender

Gender

- Men
- Women
- Unisex

Shop By Price

- Under ₹ 2,500.00
- ₹ 2,501.00 - ₹ 7,500.00

Men's Shoes
Nike Air Force 1 Mid '07
White
MRP: ₹ 10795

Men's Shoes
Nike Dunk Low Retro SE
Beige
MRP: ₹ 9695

Men's Short-Sleeve Graphic Fitness Top
Nike Dri-FIT UV Hyverse
Teal
MRP: ₹ 2495

Women's Shoes
Nike Waffle One SE
Pink
MRP: ₹ 7895

Men's Shoes
Nike Court Vision Low Next Nature
Black
MRP: ₹ 4995

Men's Shoes
Nike Air Force 1 React
White
MRP: ₹ 13295

Women's Shoes
Air Jordan 1 Elevate Low
White
MRP: ₹ 11895

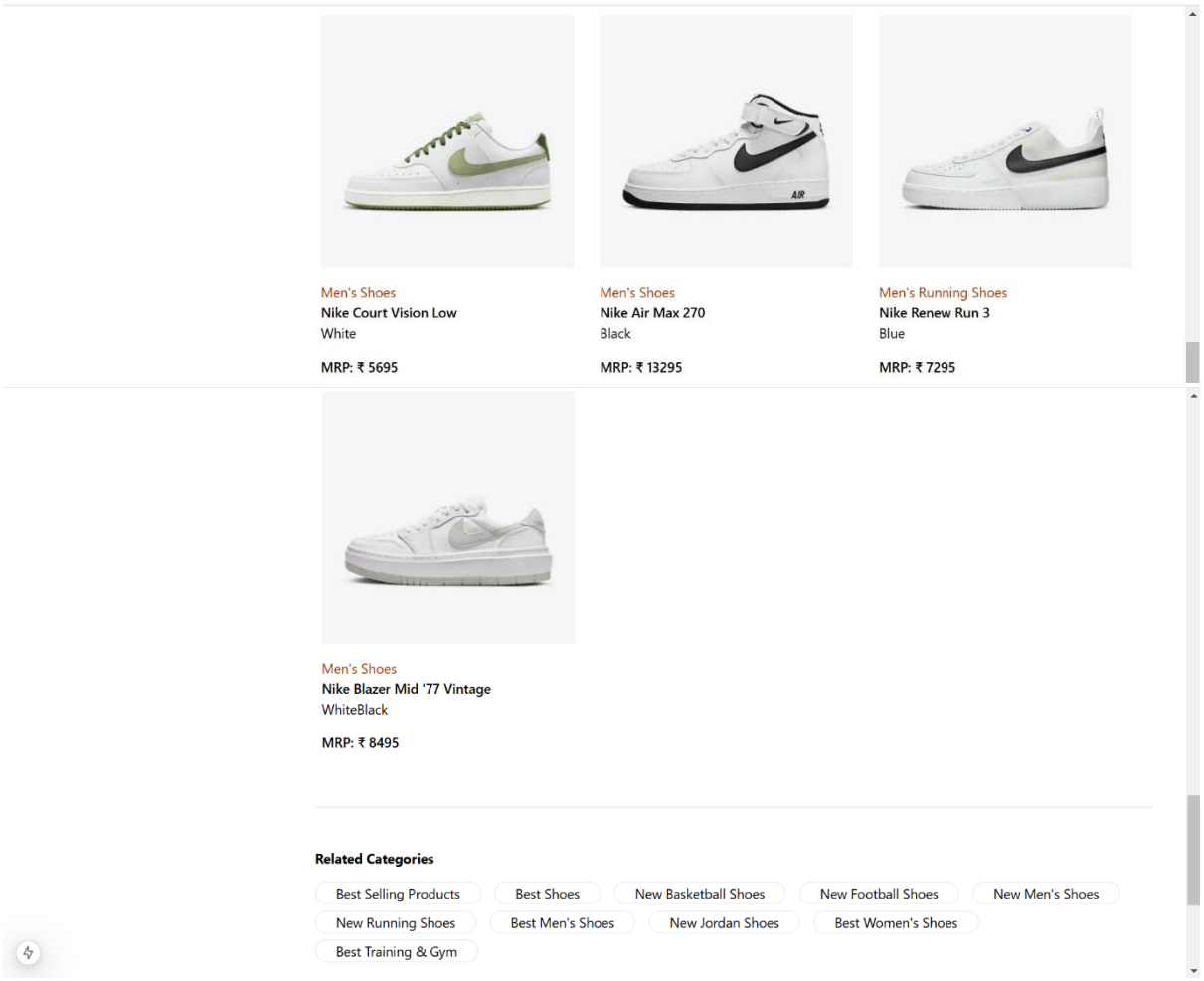
Men's Running Shoes
Nike Pegasus 40
Gray
MRP: ₹ 9795

Men's Running Shoes
Nike Zoom Fly 5
BlackWhite
MRP: ₹ 11295

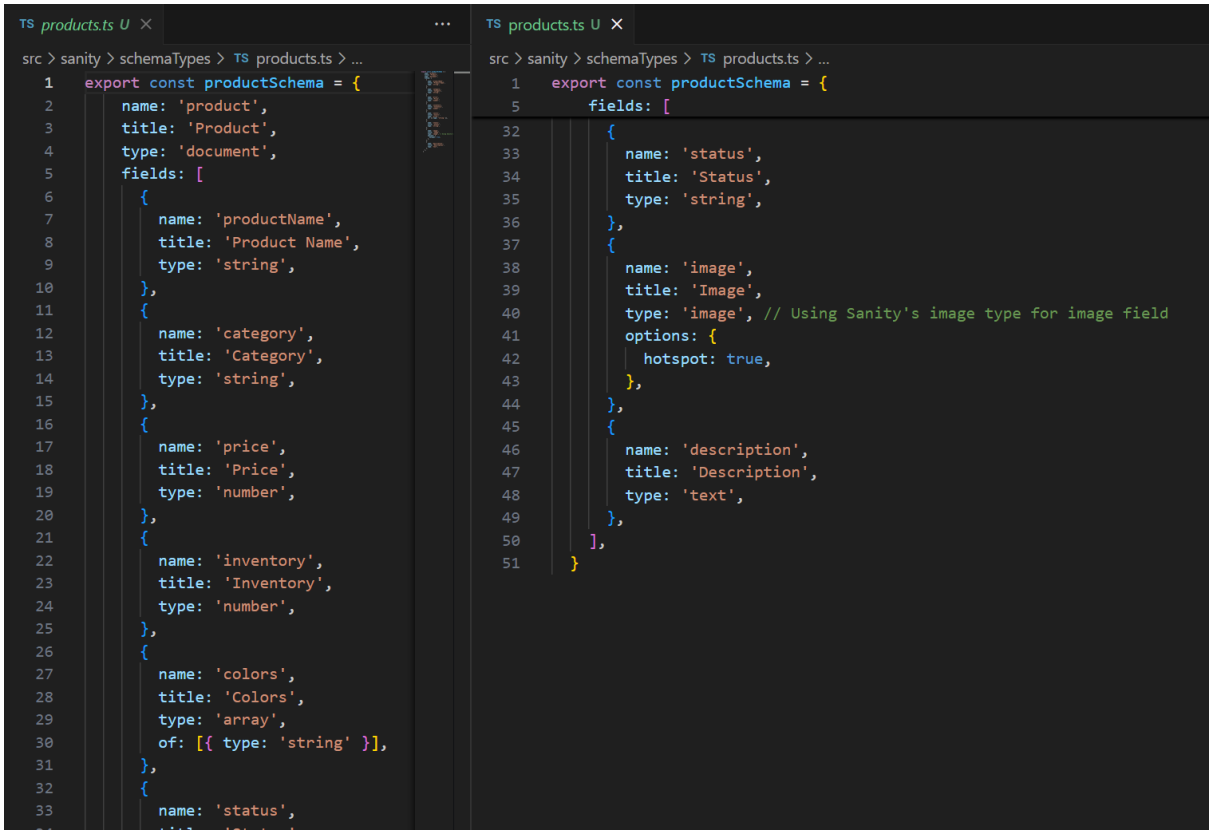
Men's Training Shoes
Nike Metcon 8
Black
MRP: ₹ 10295

Women's Shoes
Nike Air Force 1 PLT.AF.ORM
White
MRP: ₹ 8695

Women's Basketball Jersey
Nike Standard Issue Basketball Jersey
White
MRP: ₹ 2895



3. **Schema Enhancements in Sanity CMS:**
   - Refined the product schema to meet the application's evolving requirements, incorporating new fields and validations.
   - **Key Updates:**
     - **Inventory**: Added to monitor product stock levels.
     - **Colors**: Updated to store available color options as an array.



products.ts U

src > sanity > schemaTypes > products.ts > ...

```ts
export const productSchema = {
  name: 'product',
  title: 'Product',
  type: 'document',
  fields: [
    {
      name: 'productName',
      title: 'Product Name',
      type: 'string',
    },
    {
      name: 'category',
      title: 'Category',
      type: 'string',
    },
    {
      name: 'price',
      title: 'Price',
      type: 'number',
    },
    {
      name: 'inventory',
      title: 'Inventory',
      type: 'number',
    },
    {
      name: 'colors',
      title: 'Colors',
      type: 'array',
      of: [{ type: 'string' }],
    },
    {
      name: 'status',
```

products.ts U

src > sanity > schemaTypes > products.ts > ...

```ts
export const productSchema = {
    fields: [
      {
        name: 'status',
        title: 'Status',
        type: 'string',
      },
      {
        name: 'image',
        title: 'Image',
        type: 'image', // Using Sanity's image type for image field
        options: {
          hotspot: true,
        },
      },
      {
        name: 'description',
        title: 'Description',
        type: 'text',
      },
    ],
}
```