DAY 3 - API INTEGRATION AND DATA MIGRATION – General-e-commerce

1. Understanding the Provided API

- I have reviewed the API documentation for the assigned template and verified its key endpoints. The API includes an endpoint for product listings (`/products`), which aligns correctly with Template 1. The API structure and functionality match the requirements of the template.

2. Adjustments in the Schema:

- The default schemas were modified to fit the migrated data and ensure they match the API structure.

sanity > schemaTypes > TS products.ts > default

```ts
import { defineType } from "sanity"

export default defineType({
    name: 'products',
    title: 'Products',
    type: 'document',
    fields: [
        {
        name: 'name',
        title: 'Name',
        type: 'string',
        },
        {
        name: 'price',
        title: 'Price',
        type: 'number',
        },
        {
        name: 'description',
        title: 'Description',
        type: 'text',
        },
        {
        name: 'image',
        title: 'Image',
        type: 'image',
        },
        {
            name:"category",
            title:"Category",
            type: 'string',
            options:{
                list:[
                    {title: 'T-Shirt', value: 'tshirt'},
                    {title: 'Short', value: 'short'},
                    {title: 'Jeans', value: 'jeans'} ,
                    {title: 'Hoddie', value: 'hoodie'} ,
                    {title: 'Shirt', value: 'shirt'} ,
                ]
            }
        },
```

3. Data Migration Implementation Steps

1. **Define the Product Schema:**
   - Created a `products.ts` file in the `sanityTypes` folder.
   - Pasted the schema into the file.
   - Updated `sanityTypes/index.ts` to include the `products` type.

```ts
import { type SchemaTypeDefinition } from 'sanity'
import products from './products'

export const schema: { types: SchemaTypeDefinition[] } = {
  types: [products],
}

```

2. **Set Up the Import Script:**
   - Created `scripts/importSanityData.mjs` in the root directory.
   - Pasted the provided script code into the file.

```mjs
import { createClient } from '@sanity/client';
import fetch from 'node-fetch';

const client = createClient({
  projectId: "mridm5fk",
  dataset: "production",
  useCdn: true,
  apiVersion: '2025-01-13',
  token: "skt9tTLUA9FIpXbBTwoyfTgsbkYIDRxWs8TK6n1r6bbjgNp26dVKTCUIZ51T44wiKzC3S8G7kqYgaxABhkLOXGXY3Rd4R7qbkQ6d7
});

async function uploadImageToSanity(imageUrl) {
  try {
    console.log(`Uploading image: ${imageUrl}`);

    const response = await fetch(imageUrl);
    if (!response.ok) {
      throw new Error(`Failed to fetch image: ${imageUrl}`);
    }

    const buffer = await response.arrayBuffer();
    const bufferImage = Buffer.from(buffer);

    const asset = await client.assets.upload('image', bufferImage, {
      filename: imageUrl.split('/').pop(),
    });

    console.log(`Image uploaded successfully: ${asset._id}`);
    return asset._id;
  } catch (error) {
```

3. **Configure Package Script:**
   - Updated `package.json` by adding the following script under `"scripts"`:

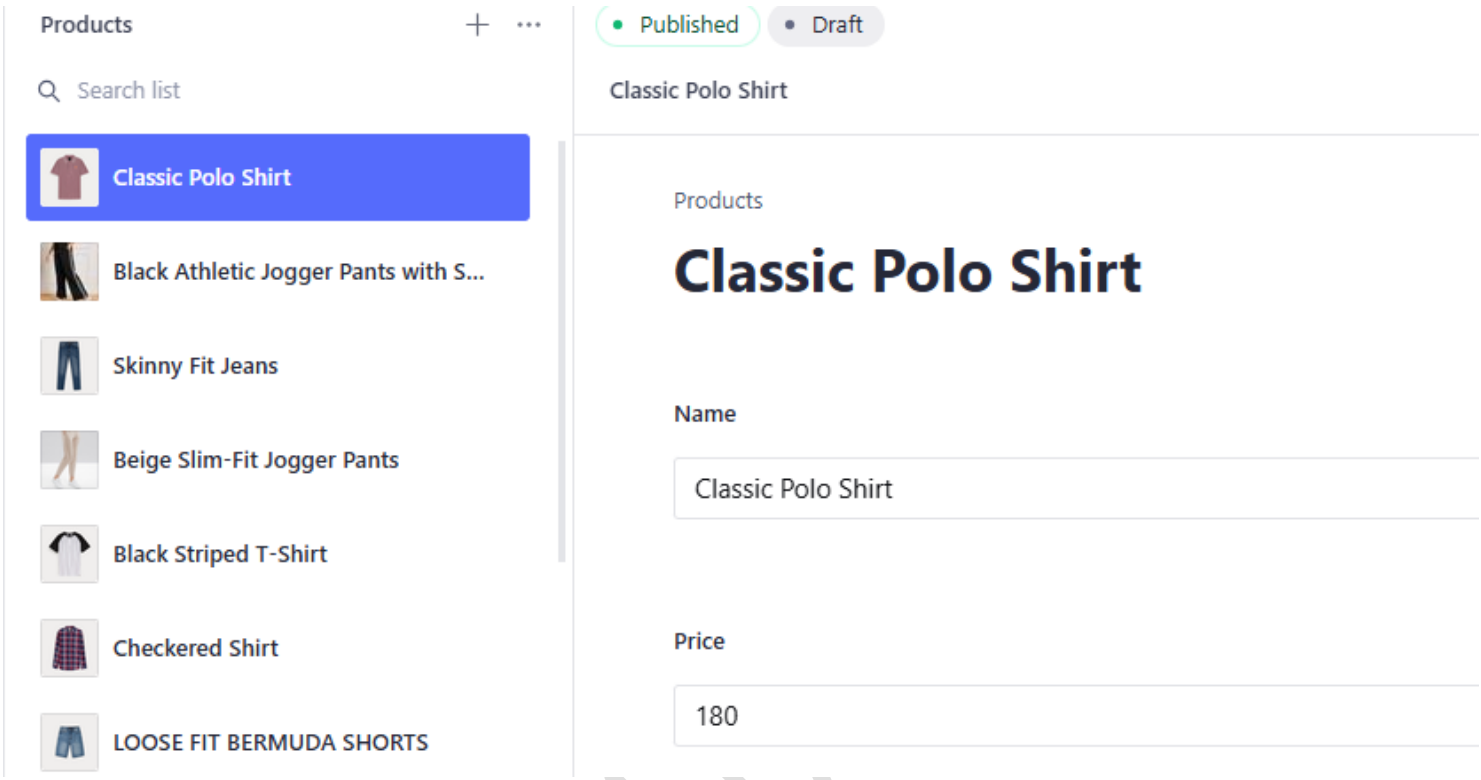
```
json
CopyEdit
"import-data": "node scripts/importSanityData.mjs"
```

```
"scripts": {
  "dev": "next dev",
  "build": "next build",
  "start": "next start",
  "lint": "next lint",
  "import-data": "node scripts/importSanityData.mjs"
},
```

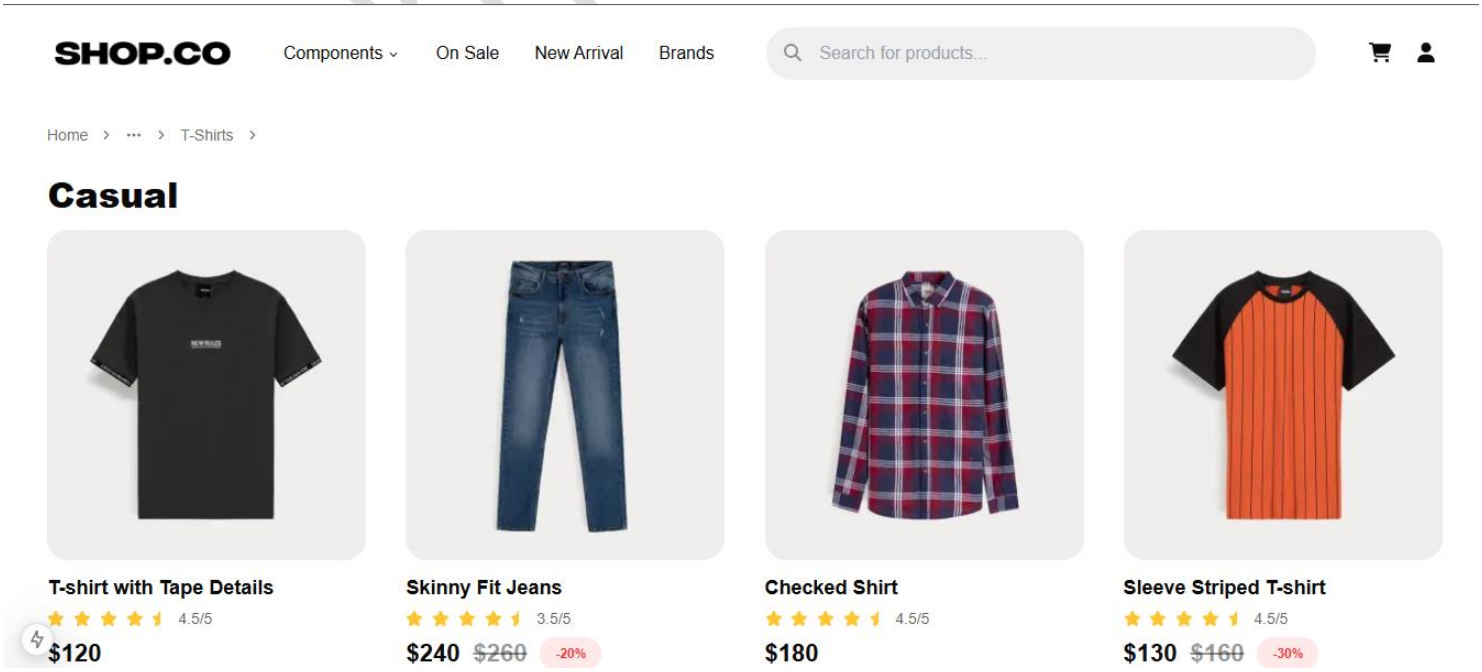
4. **Execute the Migration:**
   - Ran the script to fetch data from the API and import it into Sanity.

```
PS E:\GIAIC\e-commerce-hackathone-sanity> npm run import-data

> e-commerce-hackathone@0.1.0 import-data
> node scripts/importSanityData.mjs

Uploading image: https://cdn.sanity.io/images/7xt4qcah/production/4e2ed6a9eaa6e1413843e53f3113ccfd2104c301-278x2
Image uploaded successfully: image-4e2ed6a9eaa6e1413843e53f3113ccfd2104c301-278x296-png
Product Casual Green Bomber Jacket uploaded successfully: {
  _createdAt: '2025-02-05T12:56:34Z',
  _id: 'DGUNMUYIGLcZXFd2i1trPn',
  _rev: 'DGUNMUYIGLcZXFd2i1trMj',
  _type: 'products',
  _updatedAt: '2025-02-05T12:56:34Z',
  colors: [ 'Blue', 'Red', 'Black', 'Yellow' ],
  description: "This stylish green bomber jacket offers a sleek and modern twist on a classic design. Made from
t features snap buttons and ribbed cuffs, giving it a sporty yet refined look. The minimalist style makes it per
t-shirts or hoodies. Whether you're out with friends or just lounging, this jacket provides a laid-back yet fash
color adds a subtle, earthy tone that pairs well with a variety of outfits, making it a versatile addition to yo
  discountPercent: 20,
  image: {
    _type: 'image',
    asset: {
      ref: 'image-4e2ed6a9eaa6e1413843e53f3113ccfd2104c301-278x296-png'
```

- Finally data successfully imported in the sanity.



- Fetching Data from Sanity and display on front-end



```tsx
> const product: IProduct[] = [ ···
];

export default function Product_detail() {
  const [count, setCount] = useState(1);
  const [productData, setProductData] = useState<IProduct | null>(null);
  const params = useParams();
  const id = params?.id ? Number(params.id) : null;

  // Fetch product details from Sanity
  useEffect(() => {
    const fetchData = async () => {
      try {
        const data = await client.fetch(`*[_type == "products"]{
  _id,
  name,
  description,
    price,
    sizes,
    colors,
    image,
    discountPercent,
    isNew
}`);
        console.log("Sanity Data:", data);
      } catch (error) {
        console.error("Error fetching data:", error);
      }
    };
    fetchData();
  }, []);
```


❖ **Document Created By Anas Khan**