

DAY 3 - API INTEGRATION AND DATA MIGRATION (Fashion Product)

Objective:

The focus for Day 3 is integrating APIs and migrating data into Sanity CMS to build a functional marketplace backend. Students will learn how to utilize APIs for reference, migrate data into Sanity CMS, and ensure compatibility with their templates. This approach replicates real-world practices and prepares students to handle diverse client requirements, including integrating headless APIs or migrating existing data from popular eCommerce platforms.

Steps for Day 3:

1. Understand the Provided API:

Focused on API endpoints related to fashion Products

Key endpoints:

- **Products:** /products (fetch details for shirts, t-shirts, jeans etc)

```
template1-neon-nu.vercelapp/api/products

Pretty print

[
  {
    "discountPercent": 20,
    "sizes": [
      "S",
      "XXL",
      "XL",
      "L"
    ],
    "_id": "0dc7c847-8599-45d0-b02c-34429f7a639e",
    "name": "Casual Green Bomber Jacket",
    "price": 900,
    "imageUrl": "https://cdn.sanity.io/images/7xt4qcah/production/4e2ed6a9eaa6e1413843e53f3113ccfd2104c301-278x296.png",
    "description": "This stylish green bomber jacket offers a sleek and modern twist on a classic design. Made from soft and comfortable fabric, it features snap buttons and ribbed cuffs, giving it a sporty yet refined look. The minimalist style makes it perfect for layering over casual t-shirts or hoodies. Whether you're out with friends or just lounging, this jacket provides a laid-back yet fashionable vibe. Its muted green color adds a subtle, earthy tone that pairs well with a variety of outfits, making it a versatile addition to your casual wardrobe.",
    "category": "hoodie",
    "isNew": true,
    "colors": [
      "Blue",
      "Red",
      "Black",
      "Yellow"
    ]
  },
  {
    "description": "This white pullover hoodie offers a clean, minimalist look with its soft fabric and timeless design. The hoodie features a cozy, spacious fit, complete with an adjustable drawstring for a personalized fit and a large front pocket for added convenience. The bright white color is perfect for a fresh, casual style and pairs well with various bottoms like jeans, joggers, or shorts. Whether worn for lounging or as part of an everyday outfit, this hoodie provides comfort and versatility with a modern, relaxed flair.",
    "sizes": [
      "S",
      "XL",
      "M",
      "L"
    ],
    "_id": "17bb7c82-6fb6-462b-88c6-d39fc05fe4e2",
    "name": "Classic White Pullover Hoodie",
    "category": "hoodie",
    "discountPercent": 10,
    "isNew": false,
    "colors": [
      "Blue",
      "Red",
      "Black",
      "White"
    ]
  },
  {
    "price": 150,
    "imageUrl": "https://cdn.sanity.io/images/7xt4qcah/production/a93f8ab0d00d857ce42c057d8835e419d68a87eb-193x262.jpg"
  }
],
```

2. Validate and Adjust Your Schema

Compare your Sanity CMS schema fields with API data structure.

Adjust field names to match the imported data:

Example Adjustments:

API Field: product_title → Schema Field: name

API Field: category_name → Schema Field: category

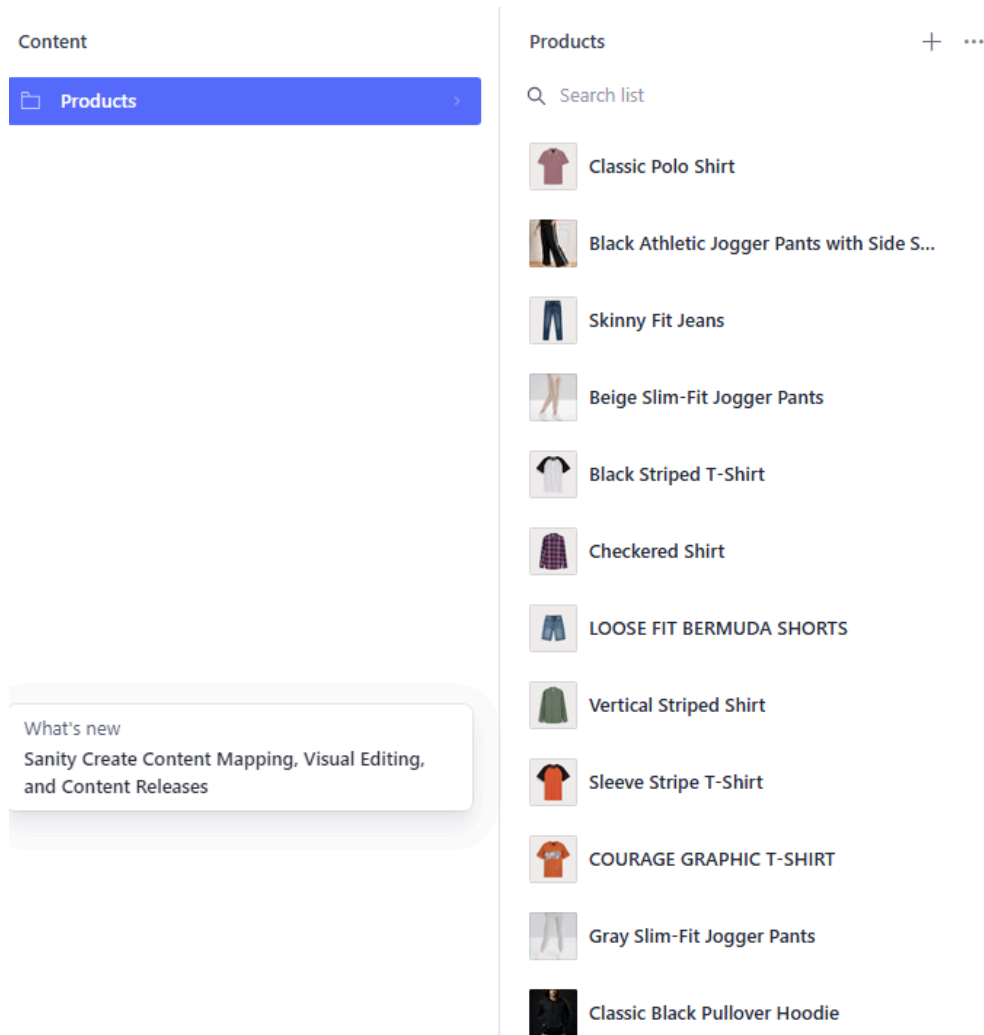
API Field: image_url → Schema Field: image

```
src> sanity > schemaTypes > TS products > [0] default > [0] fields
1  import { defineType } from "sanity"
2
3  export default defineType({
4    name: 'products',
5    title: 'Products',
6    type: 'document',
7    fields: [
8      {
9        name: 'name',
10       title: 'Name',
11       type: 'string',
12     },
13     {
14       name: 'price',
15       title: 'Price',
16       type: 'number',
17     },
18     {
19       name: 'description',
20       title: 'Description',
21       type: 'text',
22     },
23     {
24       name: 'image',
25       title: 'Image',
26       type: 'image',
27     },
28     {
29       name: 'category',
30       title: 'Category',
31       type: 'string',
32       options: {
33         list: [
34           {title: 'T-Shirt', value: 'tshirt'},
35           {title: 'Short', value: 'short'},
36           {title: 'Jeans', value: 'jeans'},
37           {title: 'Hoodie', value: 'hoodie'},
38           {title: 'Shirt', value: 'shirt'},
39         ]
36       }
37     }
38   ]
39 })
```

3. Data Migration for Fashion Products

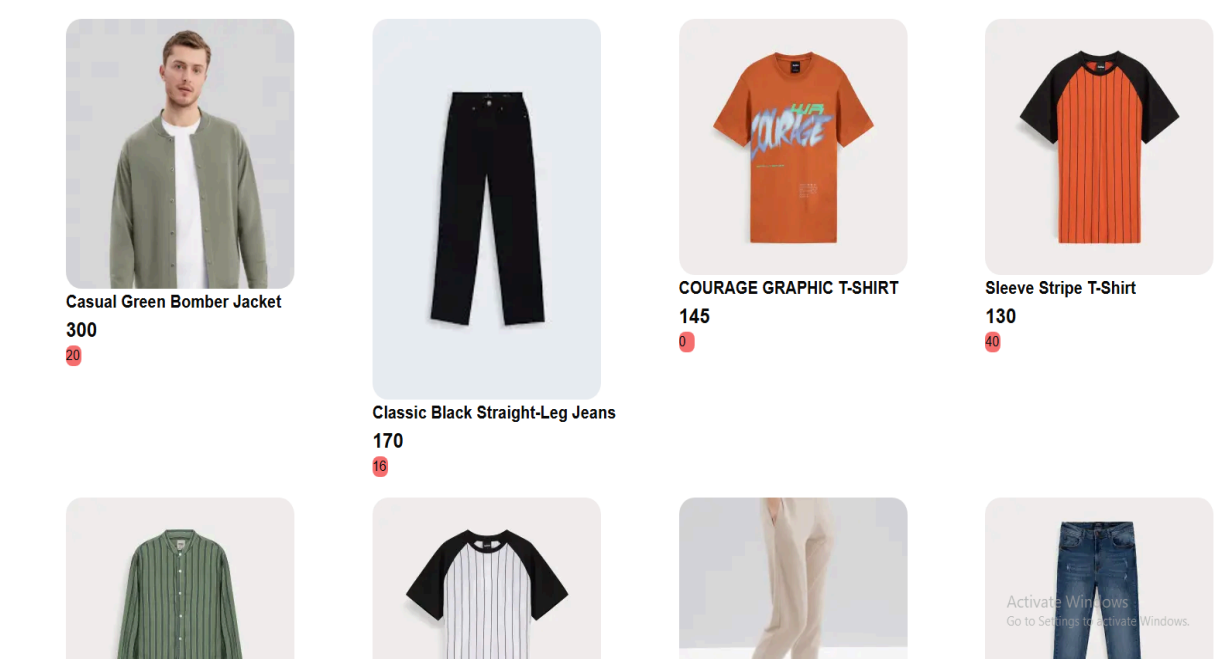
Using the Provided API

- Fetch the products from the API endpoint and transform the data.
- Rename fields (product_title → name, price → price).
- Ensure that each product includes relevant categories and images.
- Import the data into Sanity CMS.



4. API Integration in Next.js

- **Step 1: Create Utility Functions**
Write functions to fetch and transform product data dynamically.
- **Step 2: Render Data in Components**
Display the fashion products in a responsive layout.



- **Step 3: Test API Integration**

Use Postman tools to test `/api/products`.

Debug API responses to ensure the data aligns with your schema.

