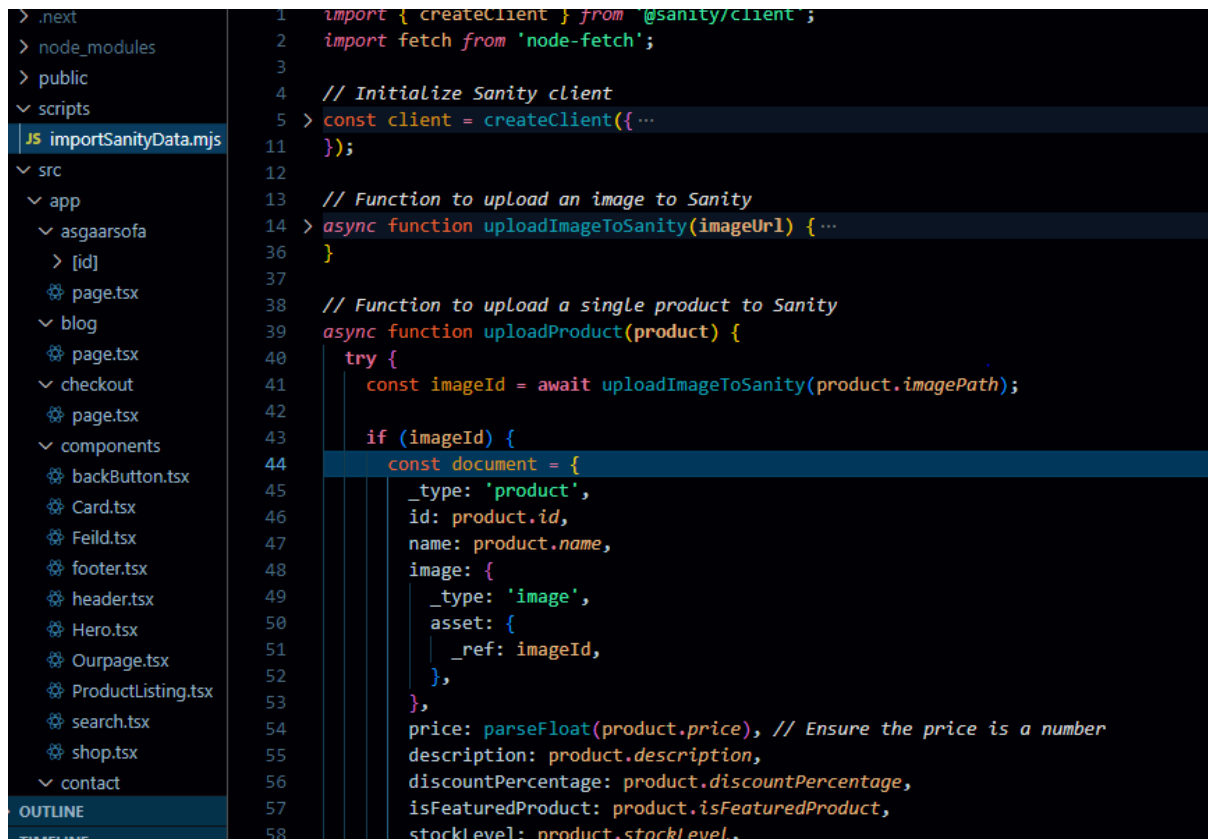# Day 3: API Integration & Data Migration for Furniture Marketplace

Custom Migration Code

```js
1   import { createClient } from '@sanity/client';
2   import fetch from 'node-fetch';
3
4   // Initialize Sanity client
5   const client = createClient({⋯
11  });
12
13  // Function to upload an image to Sanity
14  async function uploadImageToSanity(imageUrl) {⋯
36  }
37
38  // Function to upload a single product to Sanity
39  async function uploadProduct(product) {
40    try {
41      const imageId = await uploadImageToSanity(product.imagePath);
42
43      if (imageId) {
44        const document = {
45          _type: 'product',
46          id: product.id,
47          name: product.name,
48          image: {
49            _type: 'image',
50            asset: {
51              _ref: imageId,
52            },
53          },
54          price: parseFloat(product.price), // Ensure the price is a number
55          description: product.description,
56          discountPercentage: product.discountPercentage,
57          isFeaturedProduct: product.isFeaturedProduct,
58          stockLevel: product.stockLevel,
```

The custom migration code is used to transfer data from the old system to Sanity CMS.
- After validating and transforming the data, it is inserted into Sanity.
- This ensures that the data is consistent and free from duplication.

2. Sanity API Setup
- The Sanity API client is configured in the sanity/lib/client.ts file.
- Required fields:
  - projectId: Unique Sanity project ID
  - dataset: Sanity dataset (e.g., production)
  - apiVersion: API version
  - useCdn: Set to true for performance optimization
- Authentication API keys are securely stored in environment variables.
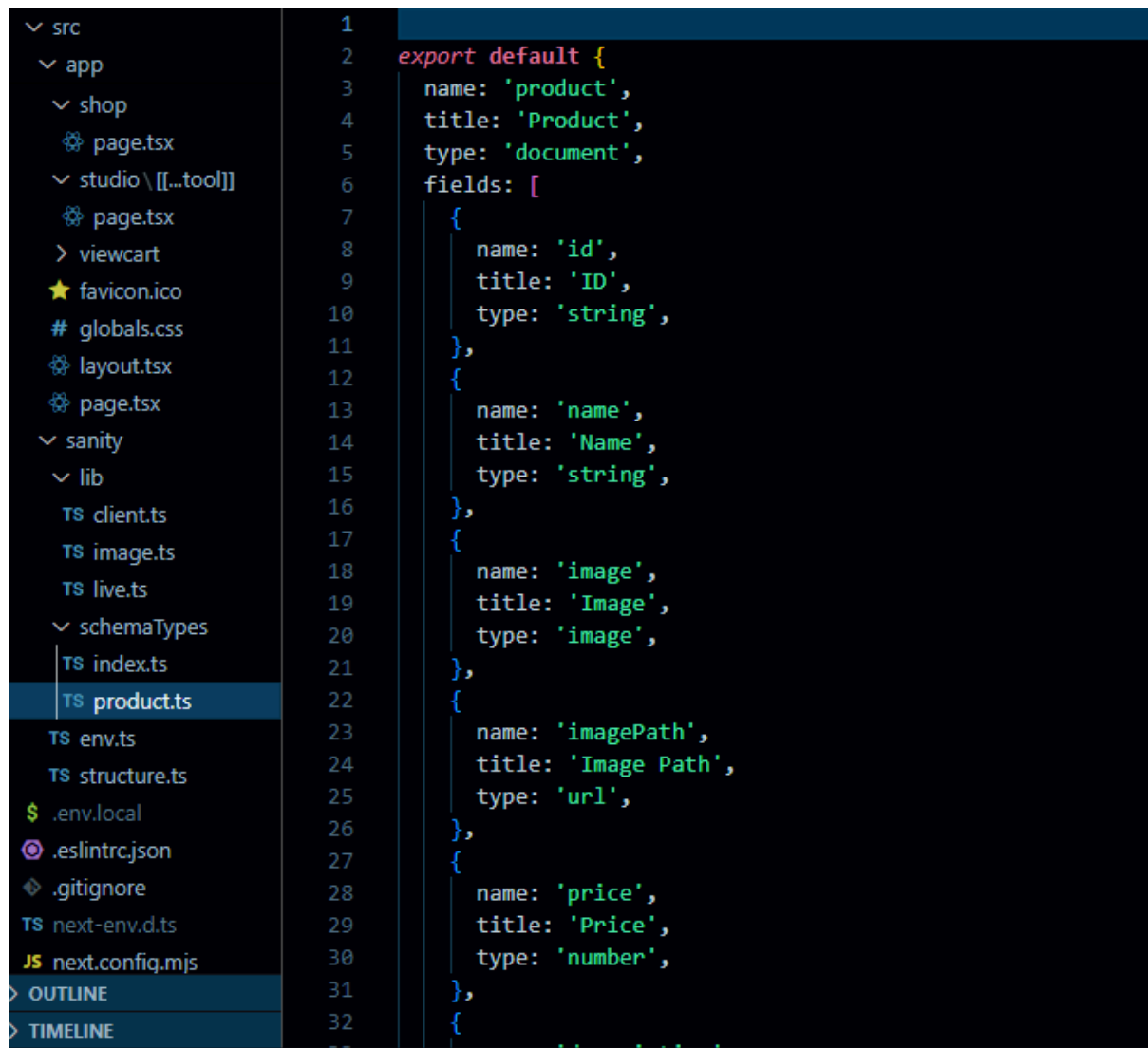
3. Client Page Code
- The client page fetches product data from Sanity.
- Asynchronous functions are used to efficiently handle API calls.

- Real-time updates are implemented to dynamically display products.


```
src
  app
    shop
      page.tsx
    studio \ [[...tool]]
      page.tsx
    viewcart
    favicon.ico
    globals.css
    layout.tsx
    page.tsx
  sanity
    lib
      client.ts
      image.ts
      live.ts
    schemaTypes
      index.ts
      product.ts
    env.ts
    structure.ts
  .env.local
  .eslintrc.json
  .gitignore
  next-env.d.ts
  next.config.mjs
```

```
1   import { createClient } from 'next-sanity'
2
3   import { apiVersion, dataset, projectId } from '../env'
4
5   export const client = createClient({
6     projectId,
7     dataset,
8     apiVersion,
9     useCdn: true, // Set to false if statically generating pages, using ISR or tag-based revalidation
10  })
11
```

Schema Code

- The schema defines the structure of data in Sanity CMS.
- It ensures data validation and consistency.

```
   1
   2   export default {
   3     name: 'product',
   4     title: 'Product',
   5     type: 'document',
   6     fields: [
   7       {
   8         name: 'id',
   9         title: 'ID',
  10         type: 'string',
  11       },
  12       {
  13         name: 'name',
  14         title: 'Name',
  15         type: 'string',
  16       },
  17       {
  18         name: 'image',
  19         title: 'Image',
  20         type: 'image',
  21       },
  22       {
  23         name: 'imagePath',
  24         title: 'Image Path',
  25         type: 'url',
  26       },
  27       {
  28         name: 'price',
  29         title: 'Price',
  30         type: 'number',
  31       },
  32       {
```
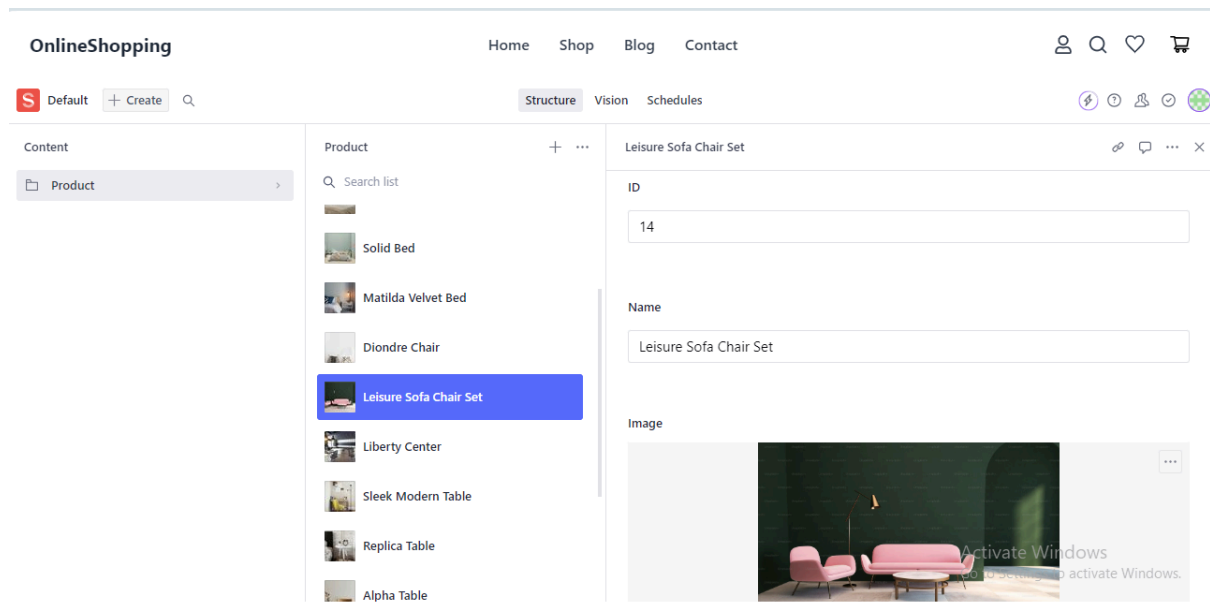
Key Fields:
- id: Unique product ID
- name: Product name
- image: Sanity image type
- price: Numeric value representing the price
- description: Short description of the product

Sanity Product Schema

The Sanity schema for furniture products is defined based on the following fields:
- id: Unique identifier
- name: Product name
- image: Sanity image object
- imagePath: URL of the product image
- price: Product price (numeric format)
- material: Material of the product (wood, metal, plastic, etc.)
- category: Product category (chair, table, sofa, etc.)