

Day 3: API Integration & Data Migration for Furniture Marketplace

Custom Migration Code

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

const __filename = fileURLToPath(import.meta.url);
const __dirname = path.dirname(__filename);
dotenv.config({ path: path.resolve(__dirname, '../.env.local') });

const client = createClient({
  projectId: process.env.NEXT_PUBLIC_SANITY_PROJECT_ID,
  dataset: process.env.NEXT_PUBLIC_SANITY_DATASET,
  useCdn: false,
  token: process.env.SANITY_API_TOKEN,
  apiVersion: '2021-08-31'
});

Tabnine | Edit | Test | Explain | Document
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('Error uploading image:', error);
  }
}
```

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.

```
import { createClient } from 'next-sanity'

import { apiVersion, dataset, projectId } from '../env'

export const client = createClient({
  projectId,
  dataset,
  apiVersion,
  useCdn: true, // Set to false if statically generating pages, using ISR or tag-based revalidation
})
```

Schema Code

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

```
import { defineType, defineField } from "sanity"

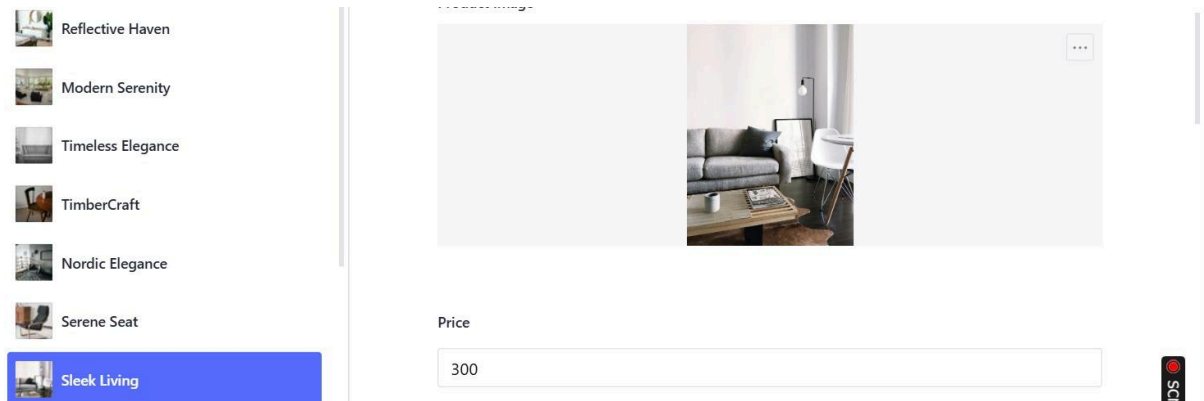
export const product = defineType({
  name: "product",
  title: "Product",
  type: "document",
  fields: [
    defineField({
      name: "category",
      title: "Category",
      type: "reference",
      to: [{
        type: "category"
      }]
    }),
    defineField({
      name: "name",
      title: "Title",
      validation: (rule) => rule.required(),
      type: "string"
    }),
    defineField({
      name: "slug",
      title: "Slug",
      validation: (rule) => rule.required(),
      type: "slug"
    }),
    defineField({
      name: "image",

```

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.)