

Day 3 – API integration {General E-Commerce}

API integration process :

Below is the step by step process of API integration

1. First of all we have opened sanity.io and logged in using google account
2. Then we created a sanity project named Nike-CloneProject, copied the installation link, and pasted it into our terminal.
3. After the installation our project is created for sanity now we are ready to use sanity.
4. We created folder and named as scripts and inside that folder we created a file named as importSanitydata.mjs in that file we paste this code:

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

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

// Create Sanity client
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'
});

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('Failed to upload image:', imageUrl, error);
  }
}
```

```

    return null;
  }
}

async function importData() {
  try {
    console.log('migrating data please wait...');

    // API endpoint containing car data
    const response = await
axios.get('https://template-03-api.vercel.app/api/products');
    const products = response.data.data;
    console.log("products ==>> ", products);

    for (const product of products) {
      let imageRef = null;
      if (product.image) {
        imageRef = await uploadImageToSanity(product.image);
      }

      const sanityProduct = {
        _type: 'product',
        productName: product.productName,
        category: product.category,
        price: product.price,
        inventory: product.inventory,
        colors: product.colors || [], // Optional, as per your schema
        status: product.status,
        description: product.description,
        image: imageRef ? {
          _type: 'image',
          asset: {
            _type: 'reference',
            _ref: imageRef,
          },
        } : undefined,
      };

      await client.create(sanityProduct);
    }

    console.log('Data migrated successfully!');
  }
}

```

```

    } catch (error) {
      console.error('Error in migrating data ==>> ', error);
    }
  }
}

const sanityProduct = {
  _type: "product",
  productName: product.productName,
  category: product.category,
  price: product.price,
  slug: { _type: "slug", current:
product.productName.toLowerCase().replace(/\s+/g, "-") },
};

await client.create(sanityProduct);

importData();

```

5. Then we opened terminal and installed axios and dotenv.
6. After the installation we created a file inside the schematype folder of the sanity and named it as product.ts in this file we pasted code provided by the link.

```

export default {
  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: "slug",
      title: "Slug",
      type: "slug",
    },
  ],
}

```

```
options: {  
  source: "productName",  
  maxLength: 200,  
},  
},  
],  
};
```

7. After successfully creating Schema of our project we locate to our sanity project in the google and we copied these all things and pasted in our .env.local file inside our project

```
NEXT_PUBLIC_SANITY_PROJECT_ID=jkz8f4e  
NEXT_PUBLIC_SANITY_DATASET=production  
NEXT_PUBLIC_SANITY_API_VERSION=2025-02-06  
SANITY_API_TOKEN=skGulnrjdmc7wa9db8y6fO5LWS2e8peD39jEvSWkESyryJIaSDMmhj  
QALvJrxpHMnbS7esE85WrBlGKgig4weFhVeCF0yPrntIV0c7q2QRv8SsvLx3sSnCLPYNBYJ  
9vMTGnxKXaIsirTLAFwc79JeJ7Xs5ffCk5G6eLhWRKPUNnQEMdGBCph
```

8. Then we run the command `npm run importSanitydata.mjs`
9. After running our sanity is deployed and we are ready to work with our code
10. Data is successfully fetching inside project.