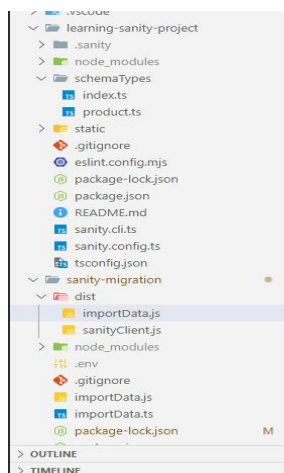


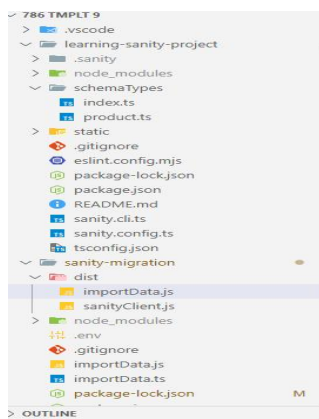
Day 3

API Integration and data Migration

API



```
1 // ...
2 var __importDefault = (this && this.__importDefault) || function (mod) {
3   return (mod && mod.__esModule) ? mod : { "default": mod };
4 };
5 Object.defineProperty(exports, "__esModule", { value: true });
6 const axios_1 = __importDefault(require("axios"));
7 const sanityClient_js_1 = require("./sanityClient.js");
8 async function uploadImageToSanity(imageUrl) {
9   try {
10     // Fetch the image from the URL and convert it to a buffer
11     const response = await axios_1.default.get(imageUrl, { responseType: 'arraybuffer' });
12     const buffer = Buffer.from(response.data);
13     // Upload the image to Sanity
14     const asset = await sanityClient_js_1.client.assets.upload('image', buffer, {
15       filename: imageUrl.split('/').pop(), // Extract the filename from URL
16     });
17     // Debugging: Log the asset returned by Sanity
18     console.log('Image uploaded successfully:', asset);
19     return asset._id; // Return the uploaded image asset reference ID
20   } catch (error) {
21     console.error('❌ Failed to upload image:', imageUrl, error);
22     throw error;
23   }
24 }
25
26 async function importData() {
27   try {
28     // Fetch data from external API
29     const response = await axios_1.default.get('https://fakestoreapi.com/products');
30     const products = response.data;
31     // Iterate over the products
32     for (const product of products) {
33       let imageRef = '';
```



```
sanity-migration > dist > importData.js > ...
26 async function importData() {
27   const sanityProduct = {
28     name: product.name,
29     ratingCount: product.rating?.count || 0,
30   };
31   // Log the product before attempting to upload it to Sanity
32   console.log('Uploading product:', sanityProduct);
33   // Import data into Sanity
34   await sanityClient_js_1.client.createOrReplace(sanityProduct);
35   console.log('✅ Imported product: ${sanityProduct.name}');
36 }
37 console.log('✅ Data import completed!');
38
39 catch (error) {
40   console.error('❌ Error importing data:', error);
41 }
42
43 importData();
44 }
```

786 TEMPLT 9

> .vscode

learning-sanity-project

> .sanity

> node_modules

schemaTypes

index.ts

product.ts

> static

.gitignore

eslint.config.mjs

package-lock.json

package.json

README.md

sanity.cli.ts

sanity.config.ts

tsconfig.json

sanity-migration

dist

importData.js

sanityClient.js

> node_modules

.env

.gitignore

importData.js

importData.ts

package-lock.json

OUTLINE

sanity-migration > dist > importData.js > ...

8 async function uploadImageToSanity(imageUrl) {

25 }

26 async function importData() {

27 try {

28 // Fetch data from external API

29 const response = await axios_1.default.get('https://fakestoreapi.com/products');

30 const products = response.data;

31 // Iterate over the products

32 for (const product of products) {

33 let imageRef = '';

34 // Upload image and get asset reference if it exists

35 if (product.image) {

36 imageRef = await uploadImageToSanity(product.image);

37 }

38 const sanityProduct = {

39 _id: `product-\${product.id}`, // Prefix the ID to ensure validity

40 _type: 'product',

41 name: product.title,

42 price: product.price,

43 discountPercentage: product.discountPercentage || 0,

44 tags: product.category ? [product.category] : [],

45 image: {

46 _type: 'image',

47 asset: {

48 _type: 'reference',

49 _ref: imageRef, // Set the correct asset reference ID

50 },

51 },

52 description: product.description,

53 rating: product.rating?.rate || 0,

54 ratingCount: product.rating?.count || 0,

55 };

Schema

> .vscode

learning-sanity-project

> .sanity

> node_modules

schemaTypes

index.ts

product.ts

> static

.gitignore

eslint.config.mjs

package-lock.json

package.json

README.md

sanity.cli.ts

sanity.config.ts

tsconfig.json

sanity-migration

dist

importData.js

sanityClient.js

> node_modules

.env

.gitignore

importData.js

importData.ts

package-lock.json

OUTLINE

TIMELINE

1 export default {

2 name: 'product',

3 type: 'document',

4 title: 'product',

5 fields: [

6 {

7 name: 'name',

8 type: 'string',

9 title: 'Product Name',

10 },

11 {

12 name: 'description',

13 type: 'string',

14 title: 'Description'

15 },

16 {

17 name: 'price',

18 type: 'number',

19 title: 'Product Price',

20 },

21 {

22 name: 'discountPercentage',

23 type: 'number',

24 title: 'Discount Percentage',

25 },

26 {

27 name: 'priceWithoutDiscount',

28 type: 'number',

29 title: 'Price Without Discount',

30 description: 'Original price before discount'

31 },

32 {

33 name: 'rating',

```
learning-sanity-project > schema types > product.ts > default > title
1  export default {
5    fields: [
42      {
43        description: 'Number of ratings'
44      },
45      {
46        name: 'tags',
47        type: 'array',
48        title: 'Tags',
49        of: [{ type: 'string' }],
50        options: {
51          layout: 'tags'
52        },
53        description: 'Add tags like "new arrival", "bestseller", etc.'
54      },
55      {
56        name: 'sizes',
57        type: 'array',
58        title: 'Sizes',
59        of: [{ type: 'string' }],
60        options: {
61          layout: 'tags'
62        },
63        description: 'Add sizes like S , M , L , XL , XXL'
64      },
65      {
66        name: 'image',
67        type: 'image',
68        title: 'Product Image',
69        options: {
70          hotspot: true // Enables cropping and focal point selection
71        }
72      }
73    ]
74  }
75 }
```

DAta Miggration

learning-sanity-project + Create

Structure Vision Schedules

Tasks

Content

product

Search list

chinez dish

New product

product

New product

Product Name

Description

Product Price

learning-sanity-project

+ Create

Q

Structure Vision Schedules

Tasks

Content

product

product

Search list

chineeZ dish

New product

Discount Percentage

Price Without Discount

Original price before discount

Rating

Rating of the product

Rating Count

learning-sanity-project

+ Create

Q

Structure Vision Schedules

Tasks

Content

product

product

Search list

chineeZ dish

New product

Tags

Add tags like "new arrival", "bestseller", etc.

Sizes

Add sizes like S , M , L , XL , XXL

Product Image

Drag or paste image here

Upload

Select

Data fetch

learning-sanity-project

+ Create

Q

Structure Vision Schedules

Tasks

DATASET

production

API VERSION

v2022-03-07

PERSPECTIVE

raw

QUERY URL [COPY TO CLIPBOARD]

https://ui3d0yi.api.sanity.io/v2022-03-07/data/query/production?query=%0A**+%5B_type+%3A

QUERY

1

2

3

4

5

6

7

8

9

10

```

* [_type == "product"]{
  product,
  description ,
}

```

PARAMS

1

2

3

```

{
}

```

RESULT

1

2

3

4

5

6

7

8

9

10

```

[ ] 1 item
0: { } 2 properties
  product: null
  description: shashilk rice

```

Fetch

Listen

Execution: 2ms

End-to-end: 193ms

Save result as

JSON

C

THE End