

## API Integration process

I add the given API to the script that works to import or fetch api data and also migrating the the products data to sanity.

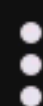
```
1      export default{
2          name: 'product',
3          type: 'document',
4          title: 'Product',
5          fields: [
6              {
7                  name: 'name',
8                  type: 'string',
9                  title: 'Name',
10                 validation: (Rule
11             },
12             {
13                 name: 'image',
14                 type: 'image',
15                 title: 'Image',
16                 options: {
17                     hotspot: true,
18                 },
19                 description: 'Up
20             },
21             {
22                 name: 'price',
23                 type: 'string',
24                 title: 'Price',
25                 validation: (Rule
26             },
27             {
28                 name: 'descripti
29                 type: 'text',
30                 title: 'Descript
31                 validation: (Rule
32                     Rule.max(150).
33             },
34             {
35                 name: 'discountP
36                 type: 'number',
37                 title: 'Discount
```

```
38  ✓  async function importData
39      try {
40          console.log('Fetchir
41
42          const response = await
43          const products = res
44
45          for (const item of p
46              console.log(`Proce
47
48          let imageRef = nul
49          if (item.imagePath
50              imageRef = await
51          }
52
53          const sanityItem =
54              _type: 'product'
55              name: item.name,
56              category: item.c
57              price: item.pric
58              description: ite
59              discountPercenta
60              stockLevel: iten
61              isFeaturedProduc
62              image: imageRef
63              ? {
64                  _type: 'in
65                  asset: {
66                      _type: '
67                      _ref: in
68                  },
69              }
70              : undefined,
71          };
72
73          console.log(`Uploa
74          const result = await
```

```
1      import { createClient }
2      import axios from 'axios'
3      import dotenv from 'dotenv'
4      import { fileURLToPath }
5      import path from 'path';
6
7      const __filename = fileURLToPath(import.meta.url);
8      const __dirname = path.dirname(__filename);
9      dotenv.config({ path: path.resolve(__dirname, '.env') });
10
11     const client = createClient({
12       projectId: "86ek1g00",
13       dataset: "production",
14       token: "sku20X0fGlo0rti",
15       apiVersion: '2025-01-15',
16       useCdn: false,
17     });
18
19     ✓ async function uploadImage(url) {
20       try {
21         console.log(`Uploading image: ${url}`);
22
23         const response = await axios.get(url);
24         const buffer = Buffer.from(response.data);
25
26         const asset = await client.assets.uploadImage(
27           __dirname,
28           filename: imageUrl,
29           buffer,
30         );
31         console.log(`Image uploaded successfully: ${asset._id}`);
32         return asset._id;
33       } catch (error) {
34         console.error('Failed to upload image:', error);
35         return null;
36       }
37     }
```



```
53         const sanityItem =
54             _type: 'product'
55             name: item.name,
56             category: item.c
57             price: item.pric
58             description: ite
59             discountPercenta
60             stockLevel: item
61             isFeaturedProduc
62             image: imageRef
63             ? {
64                 _type: 'in
65                 asset: {
66                     _type: '
67                     _ref: in
68                 },
69             }
70             : undefined,
71         };
72
73         console.log(`Uploa
74         const result = awa
75         console.log(`Uploa
76         console.log("----
77         console.log("\n\n"
78     }
79
80     console.log('Data In
81 } catch (error) {
82     console.error('Error
83 }
84 }
85
86 importData();
```



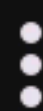
```
32         Rule.max(150).
33     },
34     {
35         name: 'discountP
36         type: 'number',
37         title: 'Discount
38         validation: (Rul
39             Rule.min(0).ma
40     },
41     {
42         name: 'isFeature
43         type: 'boolean',
44         title: 'Is Featu
45     },
46     {
47         name: 'stockLeve
48         type: 'number',
49         title: 'Stock Le
50         validation: (Rul
51     },
52     {
53         name: 'category'
54         type: 'string',
55         title: 'Category
56         options: {
57             list: [
58                 { title: 'Ch
59                 { title: 'So
60             ],
61         },
62         validation: (Rul
63     },
64 ],
65 };
```




github.com/ialias



76

 Files main ▼

Hackathone-Project- / src / sanity / lib  
/ client.ts 



ialiasgher75 2 days ago



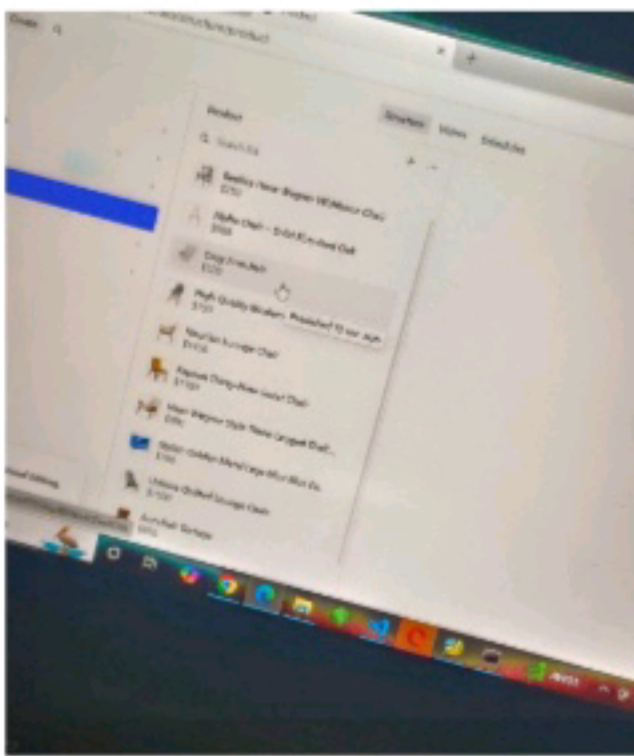
10 lines (8 loc) · 279 Bytes

Code

Blame



```
import { createClient } from 'next-  
  
import { apiVersion, dataset, proje  
  
export const client = createClient(  
  projectId,  
  dataset,  
  apiVersion,  
  useCdn: true, // Set to false if  
})
```



## Search

Search

Home

Products

Search Chair

Seating Chair - Modern White Chair  
\$120

Modern Chair - White Wood Chair  
\$100

Dining Chair  
\$120

High Quality Modern - Resin Chair  
\$120

Modern Lounge Chair  
\$120

Modern Dining Chair  
\$120

Modern Dining Chair - White Wood Chair  
\$120

Modern Dining Chair - White Wood Chair  
\$120

Modern Dining Chair - White Wood Chair  
\$120

Modern Dining Chair  
\$120

Modern Dining Chair  
\$120