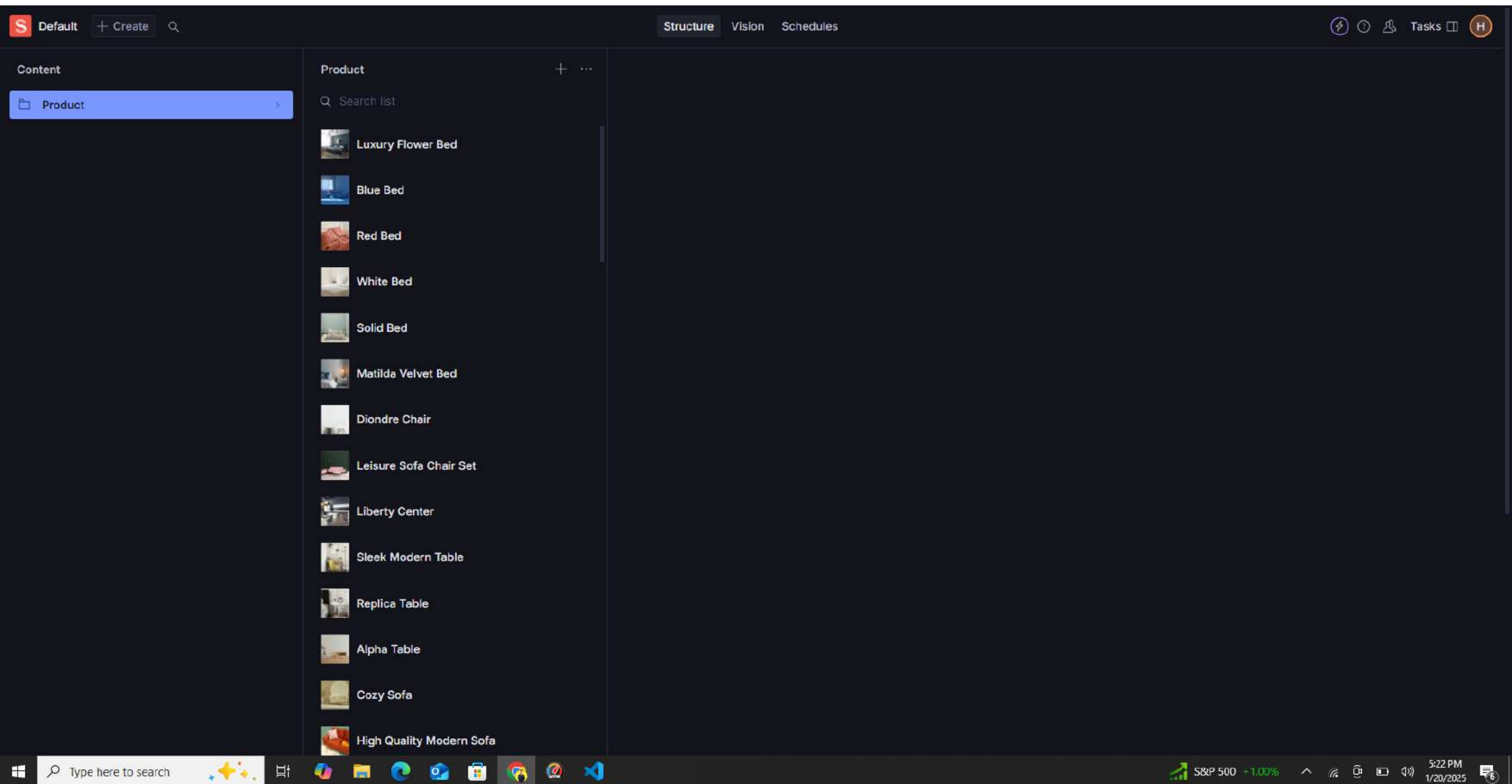


```
1 export default {
2   name: 'product',
3   title: 'Product',
4   type: 'document',
5   fields: [
6     {
7       name: 'id',
8       title: 'ID',
9       type: 'string',
10    },
11    {
12      name: 'name',
13      title: 'Name',
14      type: 'string',
15    },
16    {
17      name: 'image',
18      title: 'Image',
19      type: 'image',
20    },
21    {
22      name: 'imagePath',
23      title: 'Image Path',
24      type: 'url',
25    },
26    {
27      name: 'price',
28      title: 'Price',
29      type: 'number',
30    },
31    {
32      name: 'description',
33      title: 'Description',
34      type: 'text',
35    },
36    {
37      name: 'discountPercentage',
38      title: 'Discount Percentage',
39      type: 'number',
40    },
41    {
42      name: 'isFeaturedProduct',
43      title: 'Is Featured Product',
44      type: 'boolean',
45    },
46    {
47      name: 'stockLevel',
48      title: 'Stock Level',
49      type: 'number',
50    },
51    {
52      name: 'category',
53      title: 'Category',
54      type: 'string',
55    },
56  ],
57 };
58
```

```

1 import { createClient } from '@sanity/client';
2 import axios from 'axios';
3 import dotenv from 'dotenv';
4 import { fileURLToPath } from 'url';
5 import path from 'path';
6
7 // Load environment variables from .env.local
8 const __filename = fileURLToPath(import.meta.url);
9 const __dirname = path.dirname(__filename);
10 dotenv.config({ path: path.resolve(__dirname, '../.env.local') });
11
12 // Create Sanity client
13 const client = createClient({
14   projectId: process.env.NEXT_PUBLIC_SANITY_PROJECT_ID,
15   dataset: process.env.NEXT_PUBLIC_SANITY_DATASET,
16   useCdn: false,
17   token: process.env.SANITY_API_TOKEN,
18   apiVersion: '2021-08-31',
19 });
20
21 async function uploadImageToSanity(imageUrl) {
22   try {
23     console.log('Uploading image: ${imageUrl}');
24     const response = await axios.get(imageUrl, { responseType: 'arraybuffer' });
25     const buffer = Buffer.from(response.data);
26     const asset = await client.assets.upload('image', buffer, {
27       filename: imageUrl.split('/').pop(),
28     });
29     console.log('Image uploaded successfully: ${asset._id}');
30     return asset._id;
31   } catch (error) {
32     console.error('Failed to upload image:', imageUrl, error.message);
33     return null;
34   }
35 }
36
37 async function importData() {
38   try {
39     console.log('Migrating data, please wait...');
40
41     // Fetch products from the API
42     const response = await axios.get('https://template-0-beta.vercel.app/api/product');
43     const products = response.data;
44
45     console.log('Products fetched:', products);
46
47     for (const product of products) {
48       let imageRef = null;
49
50       if (product.imagePath) {
51         imageRef = await uploadImageToSanity(product.imagePath);
52       }
53
54       const sanityProduct = {
55         _type: 'product',
56         _id: product.id,
57         name: product.name,
58         category: product.category,
59         description: product.description,
60         discountPercentage: product.discountPercentage,
61         isFeaturedProduct: product.isFeaturedProduct,
62         stockLevel: product.stockLevel,
63         price: parseFloat(product.price),
64         image: imageRef
65       };
66
67       if (imageRef) {
68         sanityProduct.image = {
69           _type: 'image',
70           asset: {
71             _type: 'reference',
72             _ref: imageRef,
73           },
74         };
75       }
76
77       // Store original image URL
78       sanityProduct.imagePath = product.imagePath;
79
80       await client.create(sanityProduct);
81       console.log('Product created in Sanity: ${sanityProduct._id}');
82     }
83
84     console.log('Data migrated successfully!');
85   } catch (error) {
86     console.error('Error in migrating data:', error.message);
87   }
88 }
89
90 importData();

```



S

Default

+ Create

Q

Structure

Vision

Schedules

🔗

🕒

👤

Tasks

🏠

DATASET

production

API VERSION

Other

CUSTOM API VERSION

v2025-01-19

PERSPECTIVE ⓘ

raw

QUERY URL [COPY TO CLIPBOARD]

https://949228j7.api.sanity.io/v2025-01-19/data/query/production?quei

📄

QUERY

```
1 * [ type == "product" ] {
2   category,
3   id,
4   price,
5   description,
6   stockLevel,
7   imagePath,
8   discountPercentage,
9   isFeaturedProduct,
10  name,
11  "image": image.asset._ref
12 }
```

PARAMS

```
1 {
2
3 }
```

RESULT

```
[...] 63 items
▼ 0: {} 10 properties
  category: Chair
  id: null
  stockLevel: 25
  imagePath: https://plus.unsplash.com/premium_photo-1668073439372-2ceafa1222b7?q=80&w=1887&auto=format&fit=crop&ixlib=rb-4.0.3&ixid=M3wxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8fA%3D%3D
  isFeaturedProduct: true
  name: Chair Wibe
  image: image-8e3afa71668987a0516404c4a818d9538a8da023-1887x2831-jpg
  price: 1200
  description: A sleek outdoor chair with natural wooden elements and a modern look.
  discountPercentage: 10
▼ 1: {} 10 properties
  image: image-6043c0f8a743bd0c02f98edaba88057f853a1a8c-2070x1380-jpg
  category: Table
  price: 900
  description: A sturdy oak chair with a sleek and minimalist design.
  stockLevel: 18
  imagePath: https://plus.unsplash.com/premium_photo-1681412205470-77848a519359?q=80&w=2070&auto=format&fit=crop&ixlib=rb-4.0.3&ixid=M3wxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8fA%3D%3D
  discountPercentage: 10
  isFeaturedProduct: false
  name: Alpha Table
  id: null
▼ 2: {} 10 properties
  category: Table
```

▶ Fetch

▶ Listen

Execution: 13ms End-to-end: 472ms

Save result as

JSON

CSV

```

1 import React from 'react';
2 import Image from 'next/image';
3 import Link from 'next/link';
4 import ShopLine from './components/shop';
5 import Field from './components/field';
6 import Header from './components/header';
7 import Page from './components/ourpage';
8 import { client } from '@sanity/lib/client';
9 import { urlFor } from '@sanity/lib/image';
10
11 interface Product {
12   category: string;
13   id: string;
14   price: number;
15   description: string;
16   stockLevel: number;
17   imagePath: string;
18   discountPercentage: number;
19   isFeaturedProduct: number;
20   name: string;
21   image: any;
22 }
23
24 // fetch products from sanity
25 async function fetchProducts(): Promise[] {
26   const query = `*[_type == "product"]{
27     category,
28     id,
29     price,
30     description,
31     stockLevel,
32     imagePath,
33     discountPercentage,
34     isFeaturedProduct,
35     name,
36     "image":image.asset._ref
37   }`;
38   const products = await client.fetch(query);
39   return products;
40 }
41
42 const Shop = async () => {
43   const products = await fetchProducts();
44
45   return (
46     <div className="bg-gray-50">
47       <div className="mx-auto screen-2xl container mx-auto px-8 py-4">
48         {/* Header Section */}
49         <div className="bg-[#f4f4f4]>
50           <header />
51         </div>
52
53         {/* Banner Section */}
54         <div className="relative text-black">
55           <image
56             src="/shop.jpeg" // Replace with the correct image file path
57             alt="Shop Banner"
58             height={400}
59             width={1600}
60             className="w-full h-40 md:h-auto object-cover"
61           />
62           <div className="absolute top-1/2 left-1/2 transform -translate-x-1/2 -translate-y-1/2 text-xl md:text-5xl font-sans font-weight-bold">
63             <Shop />
64           </div>
65
66           {/* Breakcrumb Section */}
67           <div className="absolute top-1/2 left-1/2 transform -translate-x-1/2 -translate-y-1/2 mt-34">
68             <div className="text-gray-700 text-sm md:text-xl flex items-center">
69               <div href="/">{/* className="font-weight-bold hover:underline" */}
70               <Link />
71             </div>
72             <div className="font-bold md-2">{"/"}</div>
73             <div href="/shop">{/* className="hover:underline" */}
74             <Link />
75             </div>
76             </div>
77
78           {/* Shop Line Section */}
79           <div className="my-5">
80             <ShopLine />
81             </div>
82
83           {/* Product List */}
84           <div className="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-4 gap-8">
85             {products.map((product: Product, index) => {
86               <div
87                 key={index}
88                 className="flex flex-col items-center bg-white shadow-md rounded-lg overflow-hidden transition-transform transform hover:scale-105">
89                 <div
90                   className="h-[250px] w-full object-cover"
91                 />
92                 <div
93                   className="p-4 text-center">
94                     {/* Product Name */}
95                     <div
96                       className="text-lg font-medium text-gray-800">{product.name}</div>
97                     {/* Product Price */}
98                     <div
99                       className="text-xl font-sans font-weight-bold text-gray-800 mt-2">{product.price}</div>
100                   </div>
101                 </div>
102               )}
103             </div>
104
105             {/* Additional Sections */}
106             <div className="justify-center mx-auto">
107               <Page />
108             </div>
109           </div>
110         </div>
111       </div>
112     );
113   };
114 }
115
116 export default Shop;
117
118
119
120

```





Shop

[Home](#) > [Shop](#)

Filter  

Showing 1-6 of 32 results

Show Sort by



Chair Wibe
\$1200



Alpha Table
\$900



Replica Table
\$750



Sleek Modern Table
\$2000



Liberty Center
\$1100



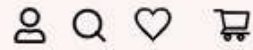
Leisure Sofa Chair Set
\$1800



Diondre Chair
\$720



Matilda Velvet Bed
\$600



Shop

Home > Shop



Filter



Showing 1-6 of 32 results

Show

16

Sort by

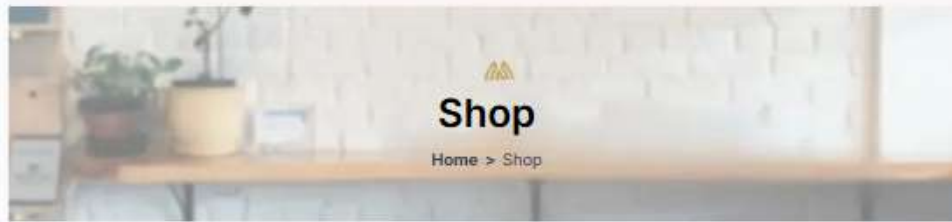
Default



Chair Wibe

\$1200





Shop

Home > Shop

Filter

Showing 1-6 of 32 results

Show 16 Sort by Default



Chair Wibe
\$1200



Alpha Table
\$900



Replica Table
\$750



Sleek Modern Table
\$2000



Liberty Center
\$1100



Leisure Sofa Chair Set
\$1800



Diondre Chair
\$720



Matilda Velvet Bed
\$600



Solid Bed
\$100



White Bed
\$120



Red Bed
\$320



Armchair Chair Set
\$850

