

'DAY 3 – API INTEGRATION – SHOP.CO'

OBJECTIVE:

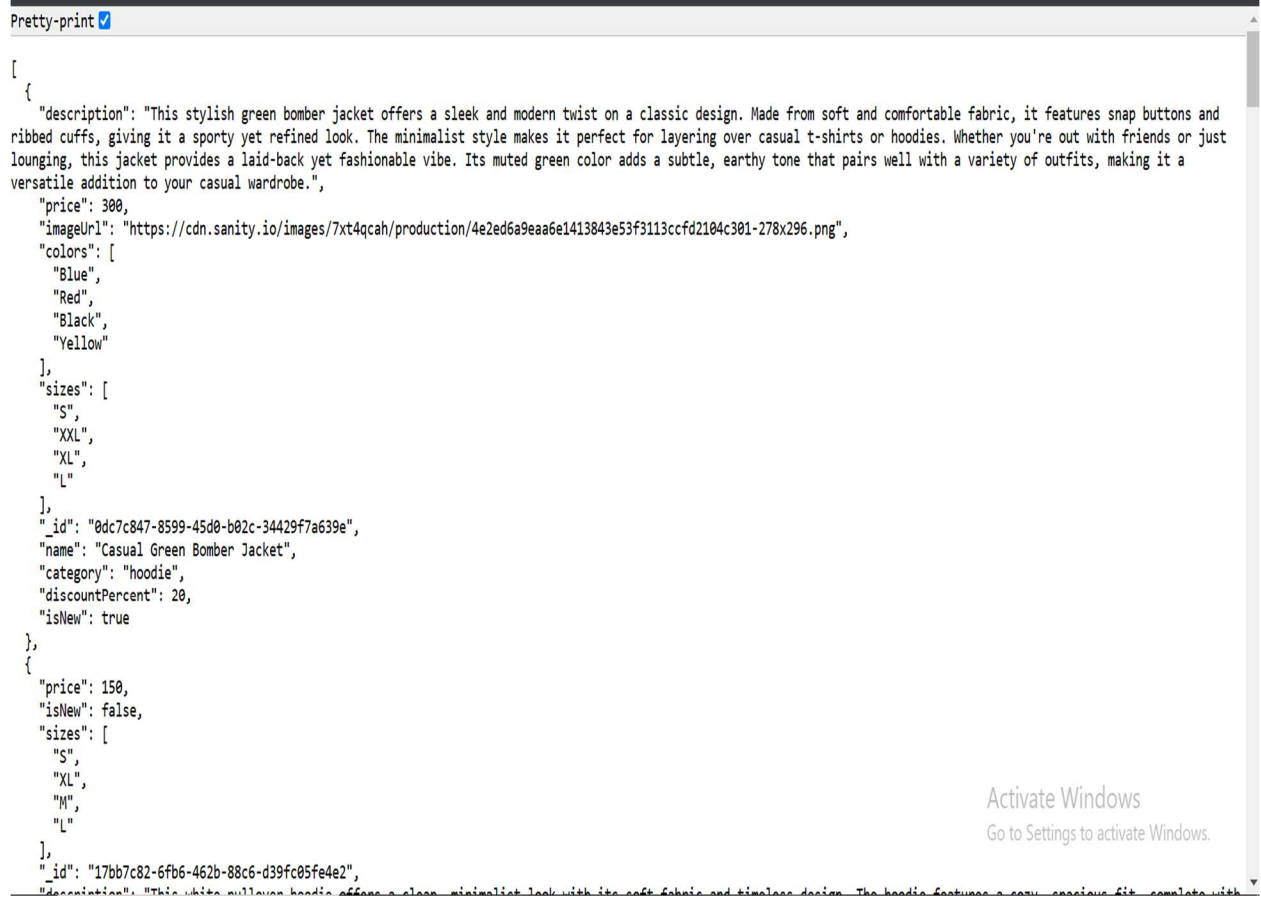
On day 3 I have migrated my data from API to my Marketplace's Sanity (CMS) and then fetches it to front end.

1.API INTEGRATION:

I have integrated the external api to my sanity.

1.1 API Testing:

At first I have Tested the API using my browser development tool.



```
[
  {
    "description": "This stylish green bomber jacket offers a sleek and modern twist on a classic design. Made from soft and comfortable fabric, it features snap buttons and ribbed cuffs, giving it a sporty yet refined look. The minimalist style makes it perfect for layering over casual t-shirts or hoodies. Whether you're out with friends or just lounging, this jacket provides a laid-back yet fashionable vibe. Its muted green color adds a subtle, earthy tone that pairs well with a variety of outfits, making it a versatile addition to your casual wardrobe.",
    "price": 300,
    "imageUrl": "https://cdn.sanity.io/images/7xt4qcah/production/4e2ed6a9eaa6e1413843e53f3113ccfd2104c301-278x296.png",
    "colors": [
      "Blue",
      "Red",
      "Black",
      "Yellow"
    ],
    "sizes": [
      "S",
      "XXL",
      "XL",
      "L"
    ],
    "_id": "0dc7c847-8599-45d0-b02c-34429f7a639e",
    "name": "Casual Green Bomber Jacket",
    "category": "hoodie",
    "discountPercent": 20,
    "isNew": true
  },
  {
    "price": 150,
    "isNew": false,
    "sizes": [
      "S",
      "XL",
      "M",
      "L"
    ],
    "_id": "17bb7c82-6fb6-462b-88c6-d39fc05fe4e2",
    "description": "This white pullover hoodie offers a clean, minimalist look with its soft fabric and timeless design. The hoodie features a cozy, spacious fit, complete with"
  }
]
```

1.2 ADJUSTMENT TO SCHEMA:

1. **COLOR:** Color was added with multiple options.
2. **SIZE:** Size was implemented with multiple options.

FINAL SCHEMA:

```
views.tsx 4  footer1.tsx  footer.tsx  page.tsx ...\cart  cart.tsx  globals.css  page.tsx ...\app  p

src > sanity > schemaTypes > product.ts > default > fields > options > list
1  import { defineType } from "sanity"
2
3  export default defineType({
4    name: 'product',
5    title: 'Product',
6    type: 'document',
7    fields: [
8      {
9        name: 'name',
10       title: 'Name',
11       type: 'string',
12     },
13     {
14       name: 'price',
15       title: 'Price',
16       type: 'number',
17     },
18     {
19       name: 'description',
20       title: 'Description',
21       type: 'text',
22     },
23     {
24       name: 'image',
25       title: 'Image',
26       type: 'image',
27     },
28     {
29       name: "category",
30       title: "Category",
31       type: 'string',
32       options: {
33         list: [
34           {title: 'T-Shirt', value: 'tshirt'},
35           {title: 'Short', value: 'short'},
36           {title: 'Jeans', value: 'jeans'},
37           {title: 'Hoddie', value: 'hoodie'},
38           {title: 'Shirt', value: 'shirt'},
39           {title: 'Casual', value: 'Casual'},
40           {title: 'Formal', value: 'Formal'},
41           {title: 'Gym', value: 'GYM'},
42           {title: 'Fashion', value: 'Fashion'}
43         ],
44       },
45     },
46   ],
47 },
48 {
49   name: "discountPercent",
50   title: "Discount Percent",
51   type: 'number',
52 },
53 ]
```

```

{
  name:"new",
  type: 'boolean',
  title:"New",
},
{
  name:"colors",
  title:"Colors",
  type: 'array',
  of:[
    {type: 'string'}
  ]
},
{
  name:"sizes",
  title:"Sizes",
  type: 'array',
  of:[
    {type: 'string'}
  ]
}
],
)

```

2. Migration steps:

- First we made a script file named “importData.mjs”
- Then we add our credentials in this file like projectId , Token , dataset etc.

```

export const client = createClient({
  projectId: "etgiegus",
  dataset: "production",
  useCdn: true,
  apiVersion: '2025-01-13',
  token:"skwp1nzp1cGcrUCJ9phPqeoHWpffVzsNApZiYss9ureSQCRaBuP3mpPQZh7Yb
});

```

- Now we will fetch the external api using fetch function.

```

async function importProducts() {
  try {
    const response = await fetch('https://template1-neon-nu.vercel.app/api/products');

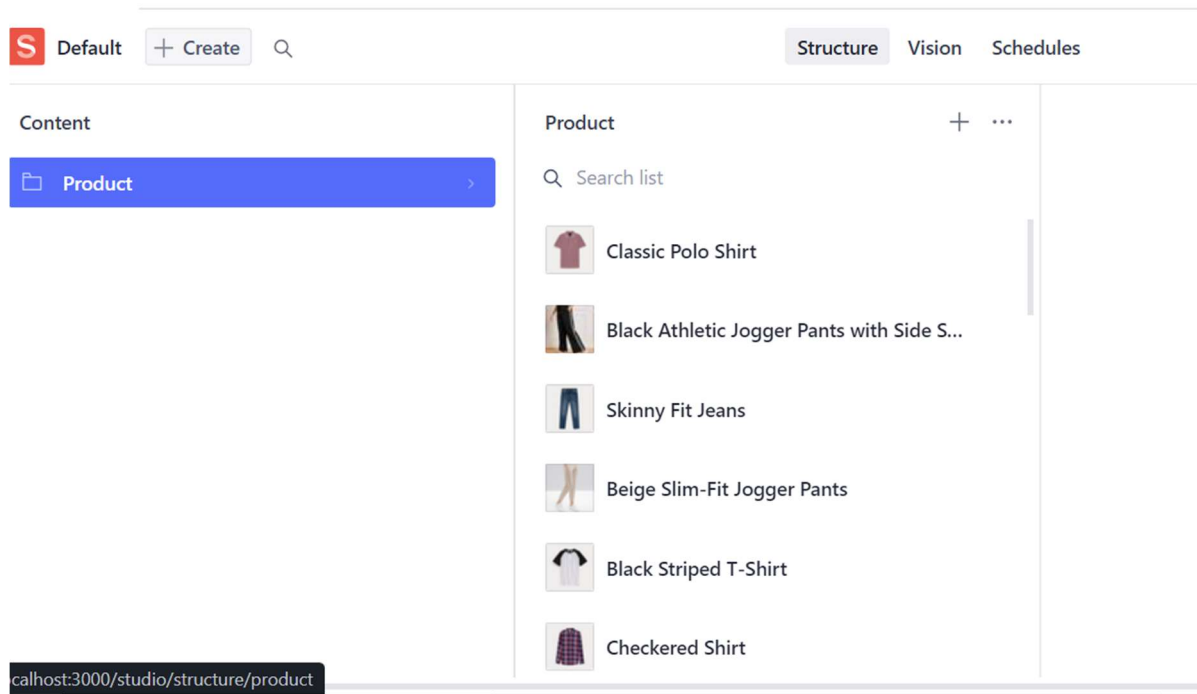
    if (!response.ok) {
      throw new Error(`HTTP error! Status: ${response.status}`);
    }
  }
}

```

- Now We will add try catch block for error handling .

```
async function importProducts() {  
  try {  
    const response = await fetch('https://template1-neon-nu.vercel.app/api/products');  
  
    if (!response.ok) {  
      throw new Error(`HTTP error! Status: ${response.status}`);  
    }  
  
    const products = await response.json();  
  
    for (const product of products) {  
      await uploadProduct(product);  
    }  
  } catch (error) {  
    console.error('Error fetching products:', error);  
  }  
}  
  
importProducts();
```

- Now finally our data is transferred to sanity .



3. FRONT END DATA FETCHING:

Now finally product data is fetched to front end. Using GROQ QUERY:

```
src > sanity > lib > TS query.ts > [🔗] allproducts
1  import { defineQuery } from "next-sanity";
2
3  export const allproducts = defineQuery(`
4    *[_type == "product" ] {
5
6      name,
7      price,
8      "imageurl": image.asset->url
9    }
10 `);
11
12
```



Classic White Pullover Hoodie

\$150



Classic Black Long Sleeve Button-Down Shirt

\$190



COURAGE GRAPHIC T-SHIRT

\$145



Vertical Striped Shirt

\$229

Activate Windows

4. CONCLUSION:

So At first we check the API and tested it using tools like POSTMAN or web developer tool and then migrated it to SANITY using migration script and then display it to frontend using GROQ query.