

Content

 **Product**

Product  

 Search list

 Rustic Vase Set 

 Zen Table

 Pure Aura

 The Lucky Lamp

 Retro Vibe

 Bed


 Marble Ease

 Product

src > sanity > schemaTypes > TS product.ts > [🔗] product > 🔑 fields

```
1  import { defineType } from "sanity"
2
3  export const product = defineType({
4    name: "product",
5    title: "Product",
6    type: "document",
7    ⚡ fields: [
8      {
9        name: "title",
10       title: "Title",
11       validation: (rule) => rule.required(),
12       type: "string"
13     },
14     {
15       name: "description",
16       type: "text",
17       validation: (rule) => rule.required(),
18       title: "Description",
19     },
20     {
21       name: "productImage",
22       type: "image",
23       validation: (rule) => rule.required(),
24       title: "Product Image"
25     },
26     {
27       name: "price",
28       type: "number",
29       validation: (rule) => rule.required(),
30       title: "Price",
```

Our Product



View Product

Share Compare Like

Bright Space
Bright Space
180



Cloud Haven Chair
Cloud Haven Chair
230



Syltherine
Syltherine
200



Bed
Bed
250



Retro Vibe
Retro Vibe
340



The Lucky Lamp
The Lucky Lamp
200



Pure Aura
Pure Aura
280



Vase Set
Vase Set
150

```

import Link from "next/link";
import Image from "next/image";
import { IoShareSocialOutline } from "react-icons/io5";
import { VscArrowSwap } from "react-icons/vsc";
import { IoHeartOutline } from "react-icons/io5";
import { useEffect, useState } from "react";
import { client } from "@sanity/lib/client";

const SecThree = () => {
  const [products, setProducts] = useState([]);
  const [loading, setLoading] = useState(true); // Loading state

  // Fetching products from Sanity
  useEffect(() => {
    const fetchProducts = async () => {
      try {
        setLoading(true); // Start loading
        const data = await client.fetch(`*[_type == "product"][0...12]{
          _id,
          title,
          description,
          price,
          "imageUrl": productImage.asset->url,
          oldPrice,
          discountTag
        }`);
        setProducts(data);
      } catch (error) {
        console.error("Error fetching products:", error);
      }
    };
  }, []);
}

```