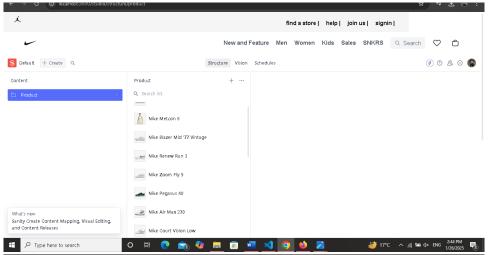
## **Day 3 API Integration**

```
import accessed (costs) from 'seartty/clear;
import accessed (costs) from 'seartty/clear;
import accessed (costs)
import access (costs)
import access (filter accessed (costs)
costs (filter accessed (costs)
costs (filter accessed (costs)
costs (costs
```

```
export const productSchema = {
    name: 'product',
    title: 'Product',
    type: 'document',
    fields: {
        f name: 'productName',
        title: 'Product Name',
        type: 'string',
        },
        f name: 'category',
        title: 'Category',
        type: 'string',
        },
        f name: 'price',
        type: 'number',
        title: 'Inventory',
        title: 'Inventory',
        title: 'Inventory',
        type: 'array',
        of: [f type: 'string' ]],
        f name: 'tatus',
        type: 'status',
        type: 'status',
        type: 'string',
        f name: 'image',
        title: 'lauge',
        type: 'tange',
        ty
```





```
app > components > shoes > @ page tox > [@] SHOES > @ useEffect() callback > @ fetchProducts > [@] fetchedProducts

import { urlFor } from "@/sanity/lib/image";

import { allFonducts, forum } from "@/sanity/lib/queries";

import { allFonducts, forum } from "@/sanity/lib/queries";

import { argument | Product | from "@/types/products";

lamport React, { useEffect, useState } from "react";

Tabmine | [Englain | Const [products, setProducts] = useState(Product[]>([]);

| Pieces Comment | Pieces Explain |
| useEffect(() => { | Pieces Comment | Pieces Explain |
| async function fetchProducts() { | const fetcheProducts: Product[] = await client.fetch(allProducts);
| setProducts(fetchedProducts); | products(); | p
```

```
| Comparison | Com
```

