

tsx M

Navbar.tsx

Product.tsx

Arrival.tsx M

page.tsx ...\app M

page.tsx ...\Sidebar M

page.tsx ...\[slug] U

page.tsx ...\Cart M

src > app > api > products > TS routes.ts > GET

```
1 import { client } from "@sanity/lib/client";
2 import { NextResponse } from "next/server";
3
4 export async function GET() {
5   const data = await client.fetch(`
6     *[_type=="products"]{
7       _id,
8       name,
9       description,
10      price,
11      "imageUrl" : image.asset->url,
12      category,
13      discountPercent,
14      "isNew": new,
15      colors,
16      sizes
17    }
18   `);
19   return NextResponse.json(data);
20 }
```


x M

Navbar.tsx

Product.tsx

Arrival.tsx M X

page.tsx ...\app M

page.tsx ...\Sideba

src > app > Components > Arrival.tsx > Arrival > calculateDiscount

```
1 import React from "react";
2 import Image from "next/image";
3 import { client } from "@sanity/lib/client";
4 import Link from "next/link";
5 import { FaStar } from "react-icons/fa";
6
7 type ArrData = {
8   id: number;
9   imageUrl: string;
10  name: string;
11  price: string;
12  priceWas: string;
13  rating: number;
14 };
15
16 async function Arrival () {
17
18   const card: ArrData[] = await client.fetch('*[_type == "products"]{_id,name,description,price,"im
19   const calculateDiscount = (price: string, priceWas: string) => {
20     if (priceWas) {
21       const discount =
22         ((parseFloat(priceWas.replace("$", "")) -
23           parseFloat(price.replace("$", ""))) /
24           parseFloat(priceWas.replace("$", ""))) *
25         100;
26       return Math.round(discount);
27     }
28     return 0;
29   };
30   return (
31     <div id="arrival">
32       { /* NEW ARRIVALS Heading */ }
33       <div className="text-center mt-12 mb-6">
34         <h1
35           className="font-IntegralCF text-4xl font-extrabold leading-[57.6px] text-center">
36           Arrival
37         </h1>
38       </div>
39     </div>
40   );
41 }
```


TS querys.ts U X

Seller.tsx M

Navbar.tsx

Product.tsx

Arrival.tsx M

src > sanity > lib > TS querys.ts > [●] products

```
1 export const products= `
2   *[_type == "products"]{
3     _id,
4     name,
5     description,
6     price,
7     "imageurl":image.asset->url,
8     category,
9     discountPercent,
10    isNew,
11    sizes,
12  }`
```

I

M

U

▼ [slug]

⚙️ page.tsx

⚙️ page.tsx

> studio

> Style

★ favicon.ico

globals.css

⚙️ layout.tsx

⚙️ page.tsx

▼ sanity

▼ lib

TS client.ts

TS image.ts

TS live.ts

TS querys.ts

▼ schemaTypes

TS index.ts

TS products.ts

TS env.ts

TS structure.ts

\$.env.local

⚙️ .eslintrc.json

📁 .gitignore

JS importData.mjs

TS next-env.d.ts

TS

main* 0 0 0

Type here to search

U

C:\Users\Kashaf Aman\Desktop\template\src\sanity\lib\live.ts

Day-3-hackathone

INSTALLATION OF NEXT.JS

Next.js Commands

Create App: `npx create-next-app@latest my-nextjs-app`

Start Dev Server: `npm run dev`

SANITY INSTALLATION COMMAND

Sanity Installation

Install CLI: `npm install -g @sanity/cli`

APIs

API Integration Process

Understand the Provided API:

Identify key endpoints:

Product Listings: e.g., `/products`

MANUALLY

Manual Import:

Next.js Commands

Create App: `npx create-next-app@latest my-nextjs-app`

Start Dev Server: `npm run dev`

SANITY INSTALLATION COMMAND

Sanity Installation

Install CLI: `npm install -g @sanity/cli`

APIs

API Integration Process

Understand the Provided API:

Identify key endpoints:

Product Listings: e.g., `/products`

MANUALLY

Manual Import:

Export data from the API or another source