DATA MIGRATION AND API INTEGRATION

Clone the migration file provide in Day 3 Hackathon Document according to your template

Migration file Link

https://github.com/developer-hammad-rehman/template1.git

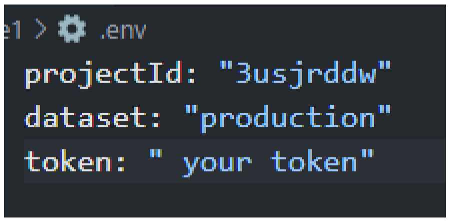
how to clone the repository?

command: **git clone https://github.com/developer-hammad-rehman/template1.git**

command: **npm install**

add .env file in your migration file

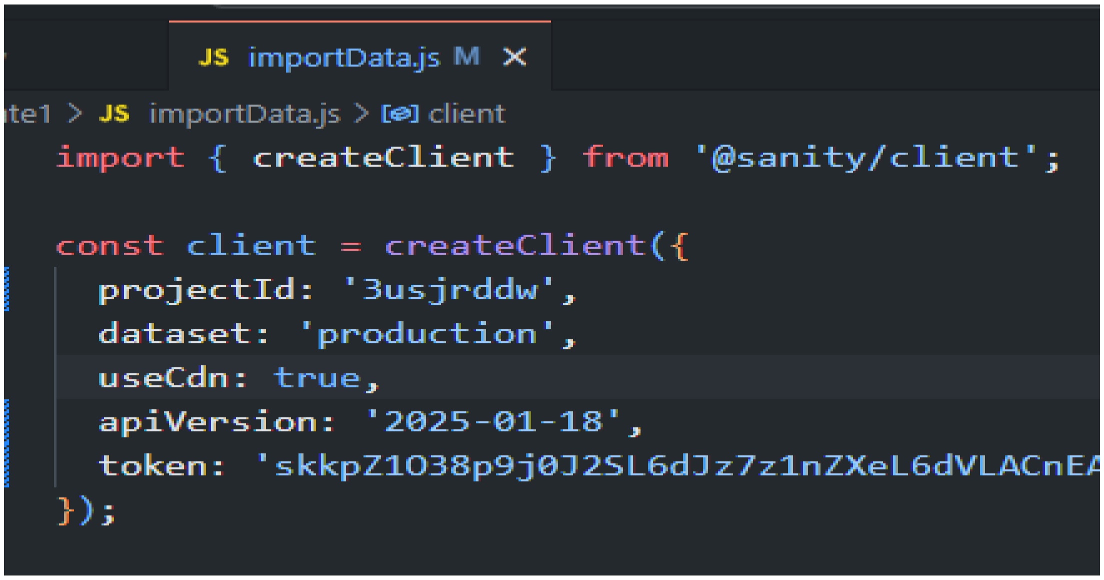
inside .env file add projectId, dataset, and token.



```
projectId: "3usjrddw"
dataset: "production"
token: " your token"
```

You have to generate token from the project of sanity which you have used in your website project.

after that, change the project id and token in importData.js file.

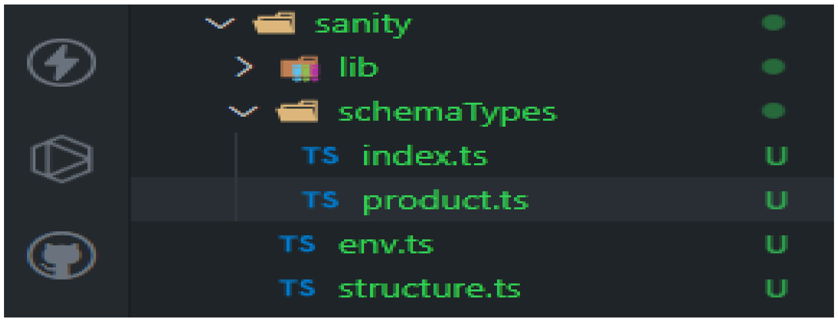


```
import { createClient } from '@sanity/client';

const client = createClient({
  projectId: '3usjrddw',
  dataset: 'production',
  useCdn: true,
  apiVersion: '2025-01-18',
  token: 'skkpZ1O38p9j0J2SL6dJz7z1nZXeL6dVLACnEA
});
```

Now your migration is connected with your main project file lets move to the main project.

Open sanity schemaTyes create file for your schema

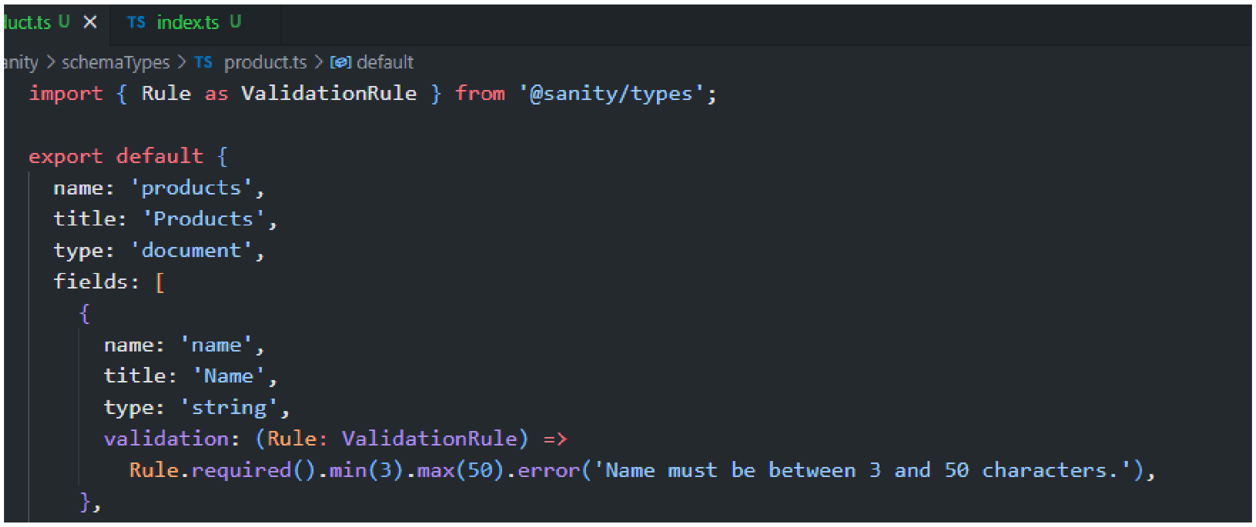
Example: product.ts



```
sanity
  lib
  schemaTypes
    index.ts        U
    product.ts      U
  env.ts            U
  structure.ts      U
```

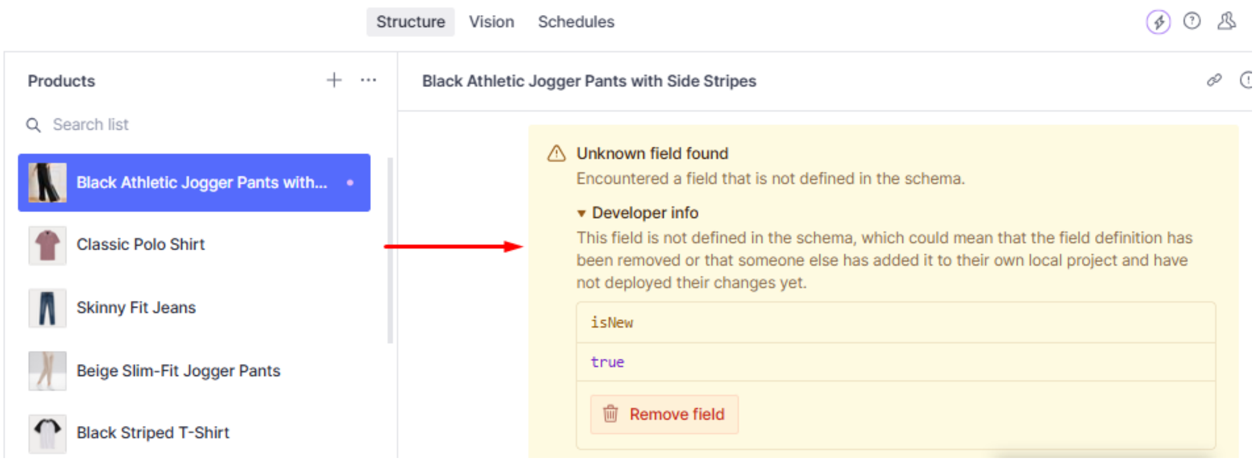
Create or paste provided schema in product.ts file.

I have added the validation in schema.



```
import { Rule as ValidationRule } from '@sanity/types';

export default {
  name: 'products',
  title: 'Products',
  type: 'document',
  fields: [
    {
      name: 'name',
      title: 'Name',
      type: 'string',
      validation: (Rule: ValidationRule) =>
        Rule.required().min(3).max(50).error('Name must be between 3 and 50 characters.'),
    },
```

I have got one error in the sanity studio

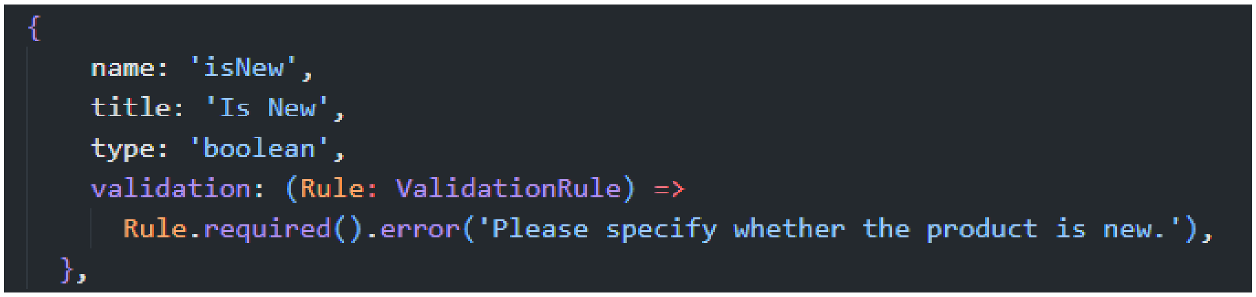


Unknown field found
Encountered a field that is not defined in the schema.

This field is not defined in the schema, which could mean that the field definition has been removed or that someone else has added it to their own local project and have not deployed their changes yet.

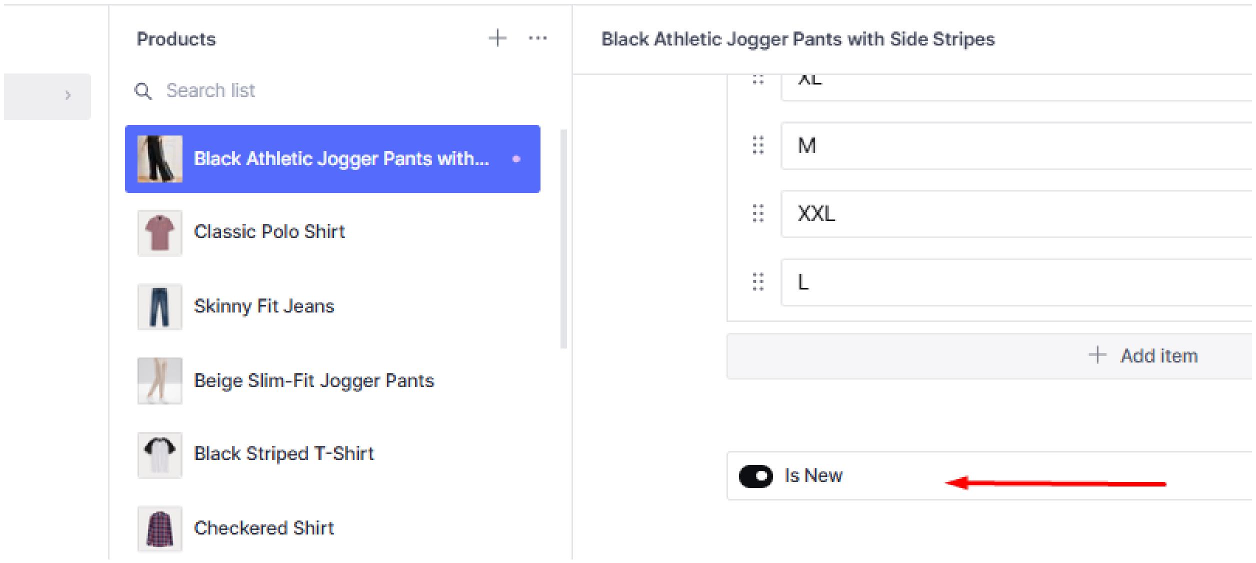
```
isNew
true
```

That I have solved by adding this code in schema.



```
{
    name: 'isNew',
    title: 'Is New',
    type: 'boolean',
    validation: (Rule: ValidationRule) =>
      Rule.required().error('Please specify whether the product is new.'),
  },
```

After solving error it shows like this.



Products

Black Athletic Jogger Pants with...

Classic Polo Shirt

Skinny Fit Jeans

Beige Slim-Fit Jogger Pants

Black Striped T-Shirt

Checkered Shirt

Black Athletic Jogger Pants with Side Stripes

XL

M

XXL

L

Is New

GROQ Query

```
*[_type == "products"] {
 _id,
 name,
 price,
 description,
 "imageUrl": image.asset->url,
 category,
 discountPercent,
 isNew,
 colors[],
 sizes[]
}
```

NOW INTEGRATE PRODUCT API AND SANITY.

Directly add the groq query into product/api/route.ts then pass the data into frontend.

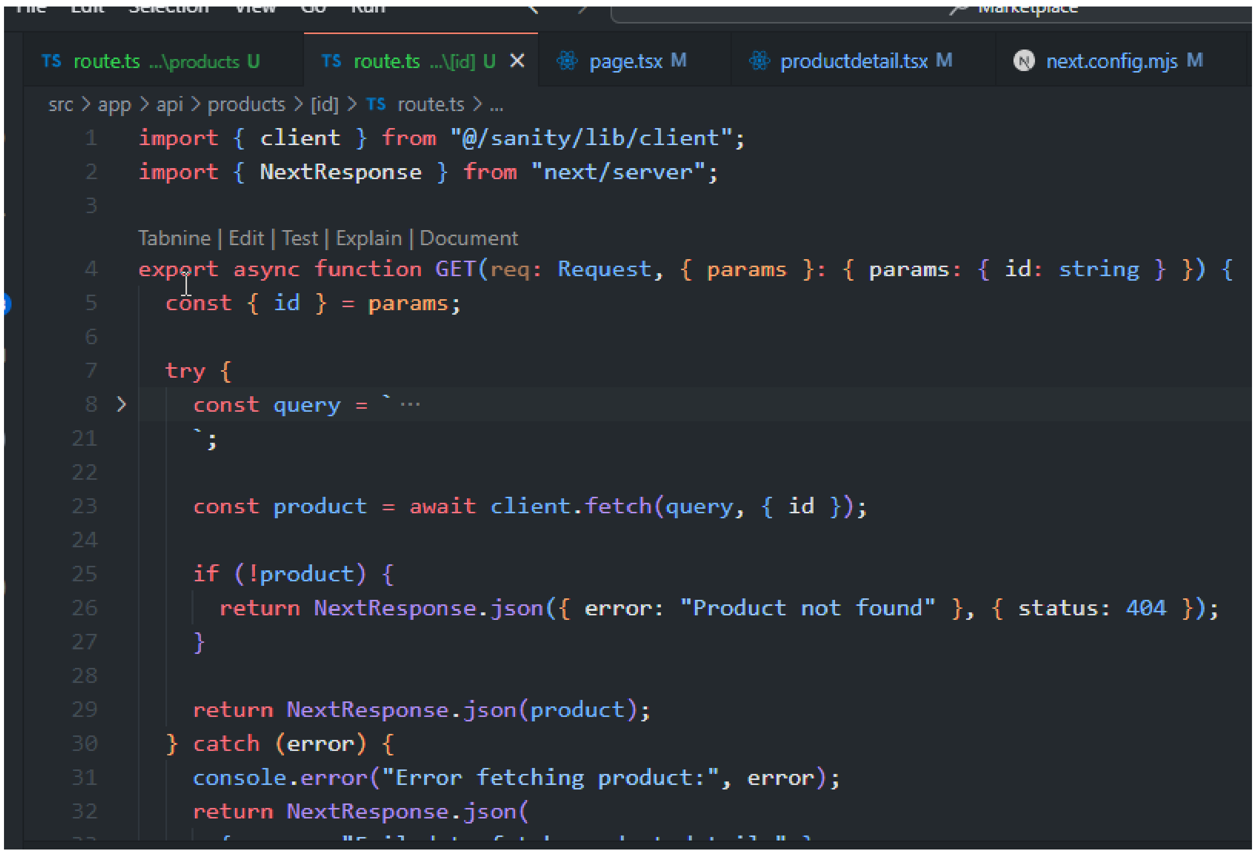
```
import { client } from "@/sanity/lib/client";
import {  NextResponse } from "next/server";

export async function GET() {
    const data = await client.fetch(`
   *[_type == "products"] {
  _id,
  name,
  price,
  description,
  "imageUrl": image.asset->url,
  category,
  discountPercent,
  isNew,
  colors[],
  sizes[]
}
    `);
    return NextResponse.json(data);
}
```

Product page.

```
// Fetch data from API
  useEffect(() => {
    const fetchProducts = async () => {
      try {
        const response = await fetch("/api/products"); // Adjust the API endpoint as per configuration
        const data = await response.json();
        setProducts(data);
        setFilteredProducts(data);
        // Set price range dynamically based on the fetched products
        const prices = data.map((product: any) => product.price);
        const minPrice = Math.min(...prices);
        const maxPrice = Math.max(...prices);
        setPriceRange([minPrice, maxPrice]);
        setSelectedPrice(maxPrice);
      } catch (error) {
        console.error("Error fetching products:", error);
      }
    };

    fetchProducts();
  }, []);
```

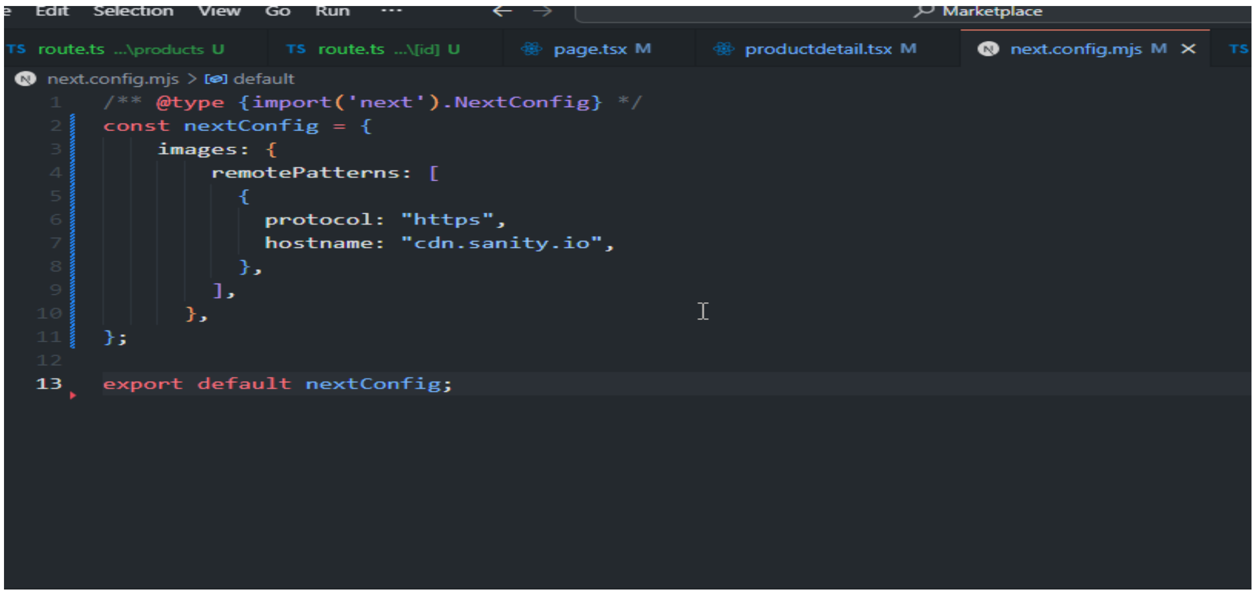
Then, create a prouct/[id]/ route.ts to get data into api from sanity through GROQ Query and then pass the data into frontend product detail page.

Product detail page api route.


TS route.ts ...\products U
TS route.ts ...\[id] U
page.tsx M
productdetail.tsx M
next.config.mjs M
src > app > api > products > [id] > TS route.ts > ...
Tabnine | Edit | Test | Explain | Document

```ts
import { client } from "@/sanity/lib/client";
import { NextResponse } from "next/server";

export async function GET(req: Request, { params }: { params: { id: string } }) {
  const { id } = params;

  try {
    const query = `···
    `;

    const product = await client.fetch(query, { id });

    if (!product) {
      return NextResponse.json({ error: "Product not found" }, { status: 404 });
    }

    return NextResponse.json(product);
  } catch (error) {
    console.error("Error fetching product:", error);
    return NextResponse.json(
```

Changes in **next.config.mjs** file for Image error



```js
/** @type {import('next').NextConfig} */
const nextConfig = {
    images: {
        remotePatterns: [
          {
            protocol: "https",
            hostname: "cdn.sanity.io",
          },
        ],
      },
};

export default nextConfig;
```