



Student Information

 Name: Rabia Bushra

 Status: Student

 Task Assigned By: Ameen Alam

 Institute: GIAIC

 Roll Number: 458649

 Email: rabiabushra94@gmail.com

Day 3

API Integration Report

Screenshots of

"Sanity CMS successfully installed using npm. Now ready to create and manage structured content for the Q-commerce marketplace!"

```
C:\Windows\System32 x + v
C:\Users\Yousuf Traders\OneDrive\Desktop\MarketPlace\Figma-assigment-template9-main>npm create sanity@latest
✓ You are logged in as rabiabushra94@gmail.com using Google
✓ Fetching existing projects

? Create a new project or select an existing one Create new project
? Your project name: day-3
Your content will be stored in a dataset that can be public or private, depending on whether you want to query your content with or without authentication.
The default dataset configuration has a public dataset named "production".
? Use the default dataset configuration? Yes
✓ Creating dataset
? Would you like to add configuration files for a Sanity project in this Next.js folder? Yes

▲
▲ It looks like you are using Next.js 15 and React 19
▲ Please read our compatibility guide.
▲ https://www.sanity.io/help/react-19
▲

? Do you want to use TypeScript? Yes
? Would you like an embedded Sanity Studio? Yes
? What route do you want to use for the Studio? /studio
? Select project template to use Clean project with no predefined schema types
? Would you like to add the project ID and dataset to your .env.local file? Yes
Added http://localhost:3000 to CORS origins
Running 'npm install --legacy-peer-deps --save @sanity/vision@3 sanity@3 @sanity/image-url@1 styled-components@6'

added 911 packages, changed 1 package, and audited 1328 packages in 5m

247 packages are looking for funding
  run `npm fund` for details

1 moderate severity vulnerability

To address all issues, run:
  npm audit fix --force

Run `npm audit` for details.

added 8 packages, and audited 1336 packages in 21s

247 packages are looking for funding
  run `npm fund` for details

1 moderate severity vulnerability

To address all issues, run:
  npm audit fix --force

Run `npm audit` for details.

Success! Your Sanity configuration files has been added to this project
C:\Users\Yousuf Traders\OneDrive\Desktop\MarketPlace\Figma-assigment-template9-main>
```

DA

Rabia Bushra

day-3

PLAN

Growth Trial

STATUS

Active

PROJECT ID

kbx260cl

📄

🚀 Getting started

🏠 Overview

👤 Members

📅 Studios

📁 Datasets

🔒 Access

↗

Webhooks

CORS origins

Tokens

y CMS CORS origin added successfully"

"Sanit

CORS origins

+ Add CORS origin

Hosts that can connect to the project API.

ORIGIN	CREDENTIALS	CREATED	
http://localhost:3000	Allowed	11 minutes	🗑
http://localhost:3333	Allowed	12 minutes	🗑

ty API token generated successfully for secure data access."

"Sani

Tokens

[+ Add API token](#)

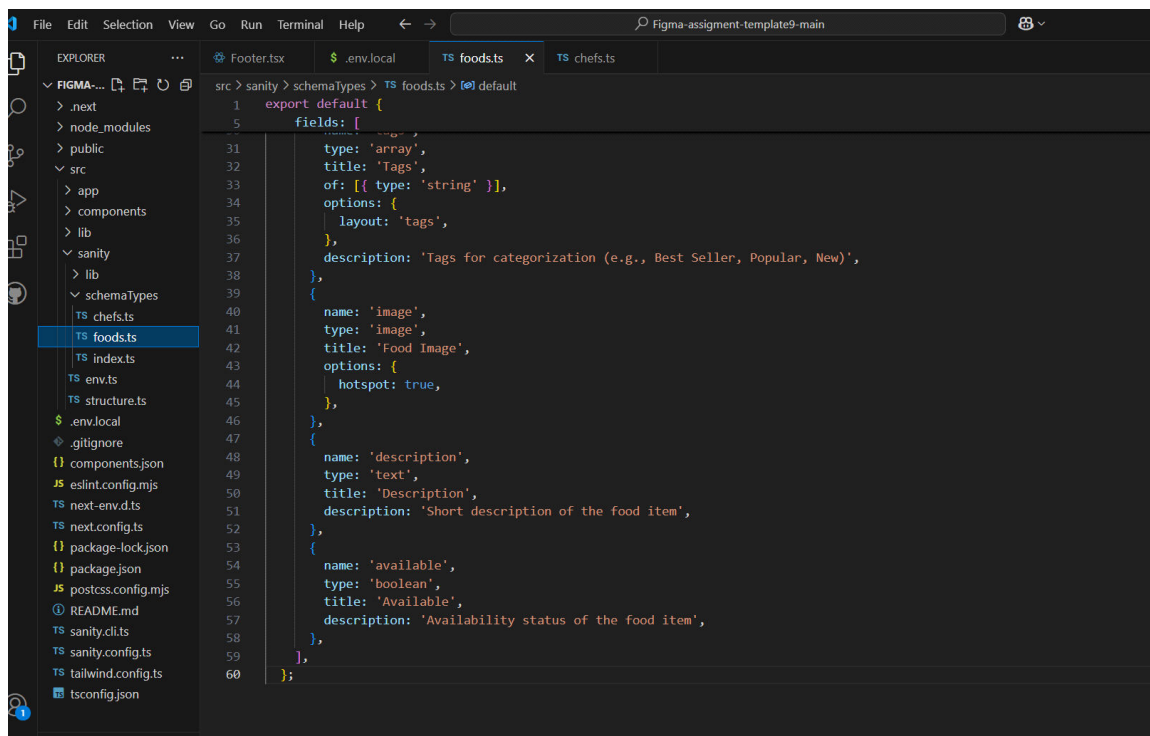
Tokens are used to authenticate apps and scripts to access project data.

NAME	PERMISSIONS	CREATED
DAY-3	Developer	0 seconds

Copy the token below – this is your only chance to do so!

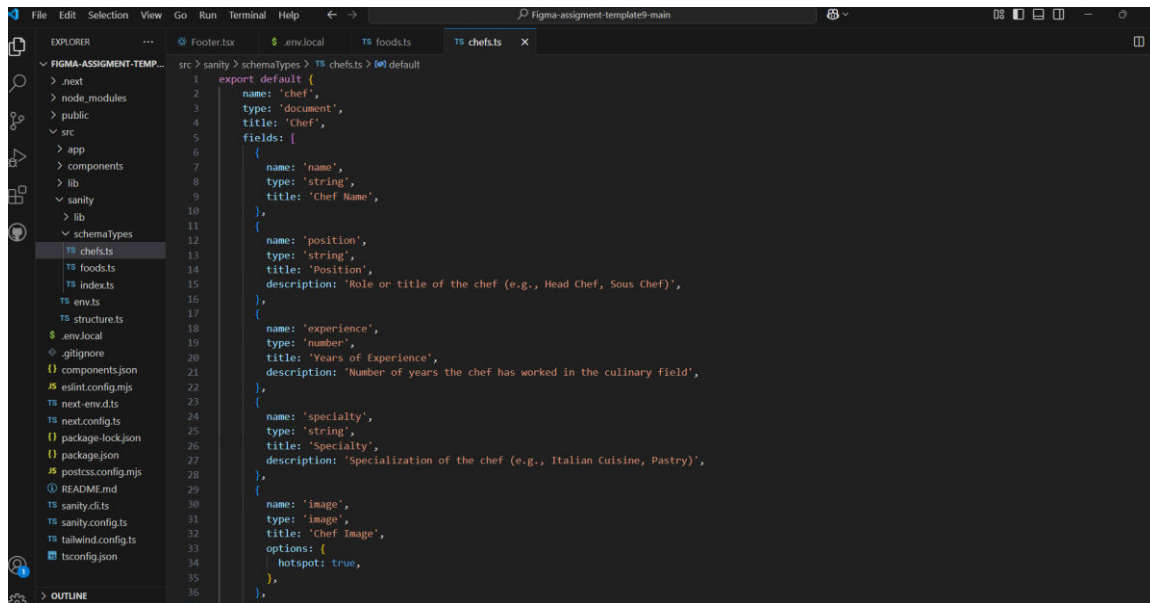
```
skc18wsM6NKU1Gn4SfyHXMB2dcVE7PD0ACgvChA2cj3VGbZjGXNMyQgvJwT8kkR
1YF3n80sKANRShgvx0GNt0CUXEs0kvFoDIo1l89Ec9PAgjlrwdAI9qxh1Gd5ZmI
vvSjGS9WLcqfdEkHv9Vf6q7C6ADZDIath6CDez04Su5n9nX5Vqt9Eh
```

"The food.ts file contains the schema for food items, including fields like name, price, category, and availability."



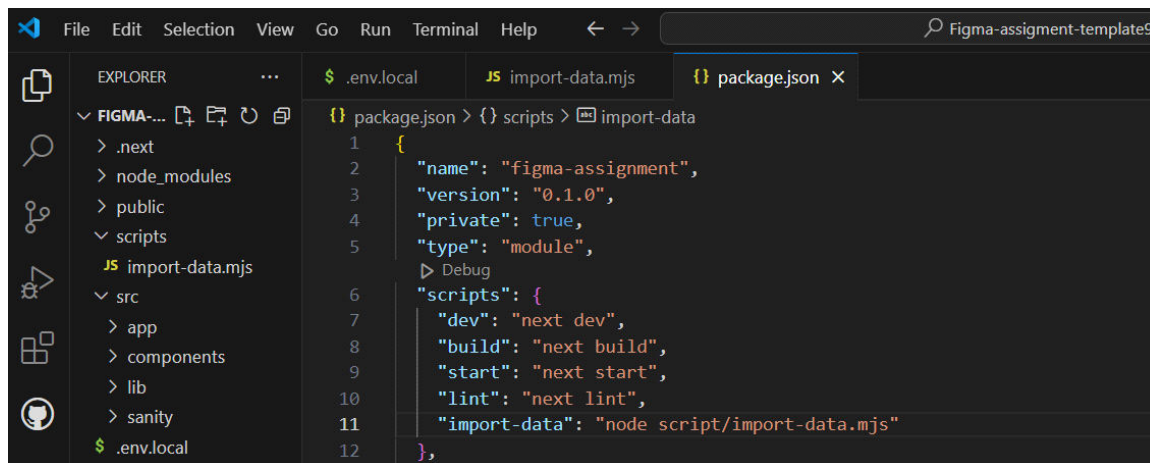
```
src > sanity > schemaTypes > TS foods.ts > default
1 export default {
5   fields: [
31     {
32       type: 'array',
33       title: 'Tags',
34       of: [{ type: 'string' }],
35       options: {
36         layout: 'tags',
37       },
38       description: 'Tags for categorization (e.g., Best Seller, Popular, New)',
39     },
40     {
41       name: 'image',
42       type: 'image',
43       title: 'Food Image',
44       options: {
45         hotspot: true,
46       },
47     },
48     {
49       name: 'description',
50       type: 'text',
51       title: 'Description',
52       description: 'Short description of the food item',
53     },
54     {
55       name: 'available',
56       type: 'boolean',
57       title: 'Available',
58       description: 'Availability status of the food item',
59     },
60   ],
61 }
```

"The chefs.ts file contains the schema for chefs, including fields like name, experience, specialty, and profile image."

A screenshot of the Visual Studio Code editor. The Explorer sidebar on the left shows a project structure with folders like .next, node_modules, public, src, app, components, lib, sanity, and schemaTypes. The schemaTypes folder is expanded, showing TS chefs.ts selected. The main editor displays the content of TS chefs.ts, which is a TypeScript interface defining a 'chef' document type. The schema includes fields for name, position, experience, specialty, and image, each with a type, title, and description. The image field also includes an options object with a hotspot property.

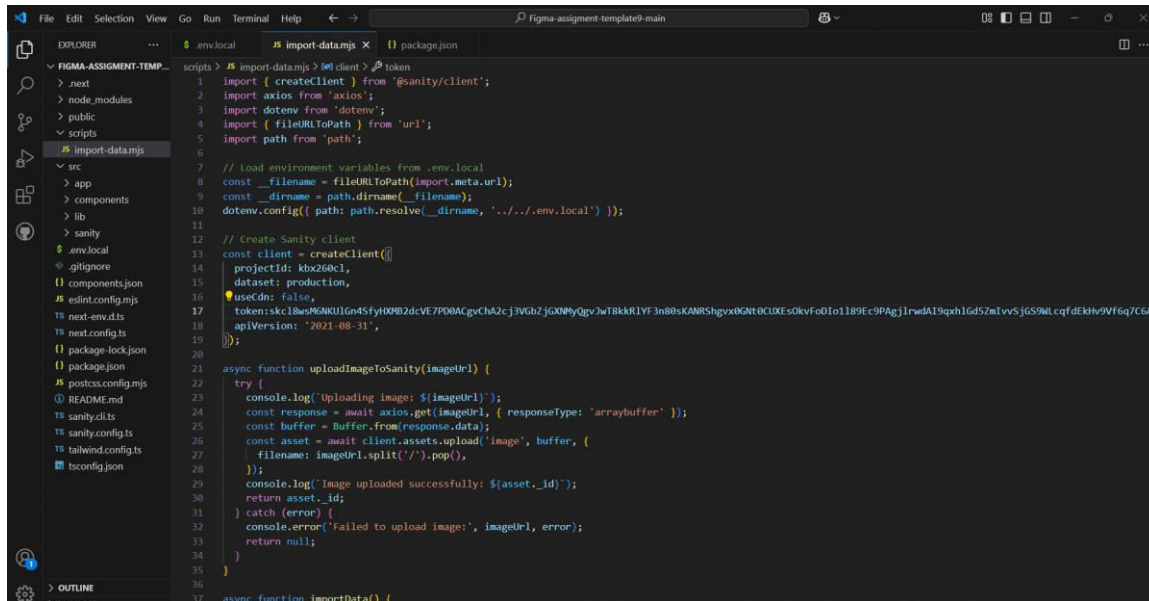
```
1 export default {
2   name: 'chef',
3   type: 'document',
4   title: 'Chef',
5   fields: [
6     {
7       name: 'name',
8       type: 'string',
9       title: 'Chef Name',
10    },
11    {
12      name: 'position',
13      type: 'string',
14      title: 'Position',
15      description: 'Role or title of the chef (e.g., Head Chef, Sous Chef)',
16    },
17    {
18      name: 'experience',
19      type: 'number',
20      title: 'Years of Experience',
21      description: 'Number of years the chef has worked in the culinary field',
22    },
23    {
24      name: 'specialty',
25      type: 'string',
26      title: 'Specialty',
27      description: 'Specialization of the chef (e.g., Italian Cuisine, Pastry)',
28    },
29    {
30      name: 'image',
31      type: 'image',
32      title: 'Chef Image',
33      options: {
34        hotspot: true,
35      },
36    },
37  ],
38 }
```

"Added `import-data` script and enabled `type: module` in `package.json` for data import and ES module support."

A screenshot of the Visual Studio Code editor. The Explorer sidebar on the left shows a project structure with folders like .next, node_modules, public, scripts, src, app, components, lib, sanity, and .env.local. The scripts folder is expanded, showing JS import-data.mjs selected. The main editor displays the content of package.json, which includes a name, version, private flag, type set to 'module', and a scripts object. The scripts object contains 'dev', 'build', 'start', 'lint', and 'import-data' scripts. The 'import-data' script is set to 'node script/import-data.mjs'.

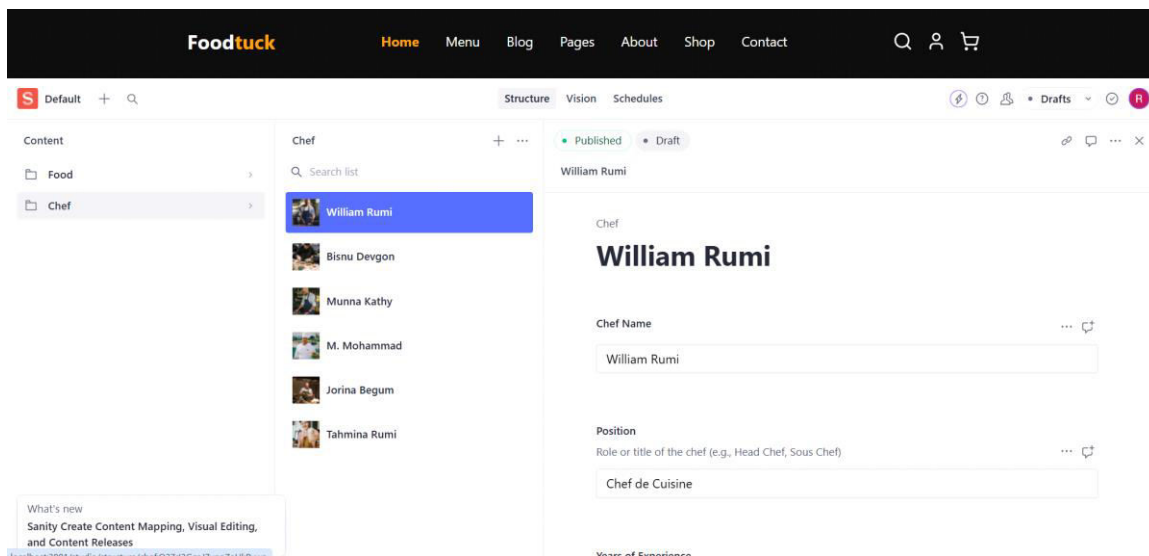
```
1 {
2   "name": "figma-assignment",
3   "version": "0.1.0",
4   "private": true,
5   "type": "module",
6   "scripts": {
7     "dev": "next dev",
8     "build": "next build",
9     "start": "next start",
10    "lint": "next lint",
11    "import-data": "node script/import-data.mjs"
12  },
13 }
```

"Added import-data.mjs for seamless data migration into Sanity CMS."

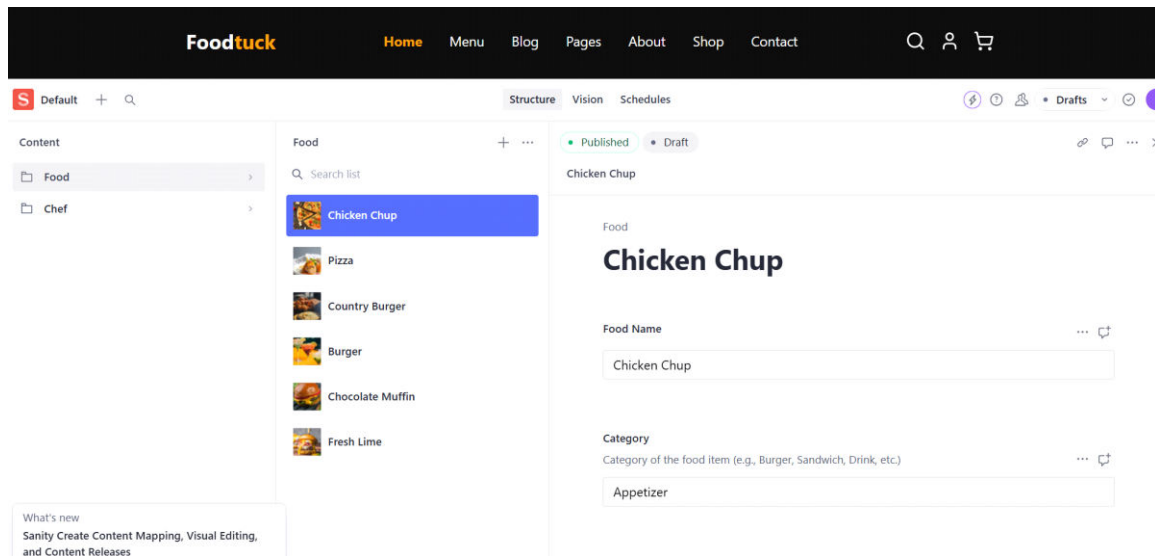


```
1 import { createClient } from '@sanity/client';
2 import axios from 'axios';
3 import dotenv from 'dotenv';
4 import { fileURLToPath } from 'url';
5 import path from 'path';
6
7 // Load environment variables from .env.local
8 const __filename = fileURLToPath(import.meta.url);
9 const __dirname = path.dirname(__filename);
10 dotenv.config({ path: path.resolve(__dirname, '../.env.local') });
11
12 // Create Sanity client
13 const client = createClient({
14   projectId: 'kx26ac1',
15   dataset: 'production',
16   useCdn: false,
17   token: 'sk1BwMGNULGn4SfyH0N82dcVE7P08ACgvChA2cj3WGb7jG0MpyQgvJwT8kkR1VE3n80sKANRShgvr0Gnt8CUDxs0kvFoDToI189Fc9Agj1rw8A19qxh1Gd5ZmIvv5jGS9Wl.cqfdeKhv9Vf6q7CGAD',
18   apiVersion: '2021-08-31',
19 });
20
21 async function uploadImageToSanity(imageUrl) {
22   try {
23     console.log('Uploading image: ', imageUrl);
24     const response = await axios.get(imageUrl, { responseType: 'arraybuffer' });
25     const buffer = Buffer.from(response.data);
26     const asset = await client.assets.upload('image', buffer, {
27       filename: imageUrl.split('/').pop(),
28     });
29     console.log('Image uploaded successfully: ', asset._id);
30     return asset._id;
31   } catch (error) {
32     console.error('Failed to upload image: ', imageUrl, error);
33     return null;
34   }
35 }
36
37 async function importData() {
```

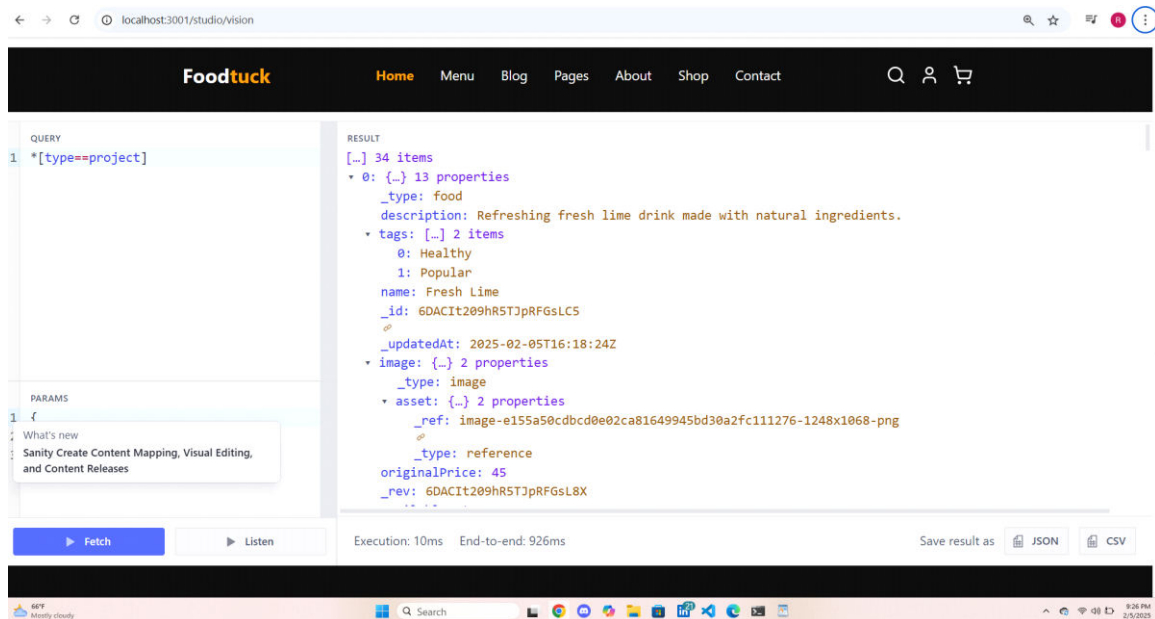
"Chefs' data successfully fetched and displayed in Sanity Studio from localhost3000/studio."



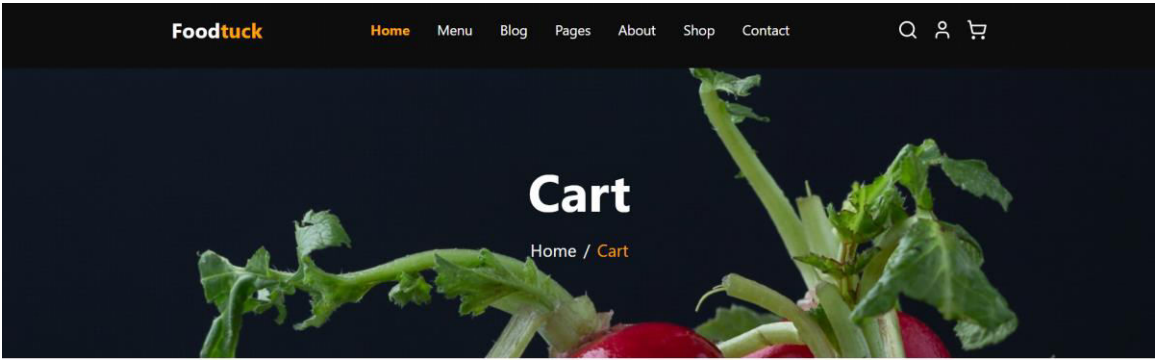
"Food data successfully fetched and displayed in Sanity Studio from localhost3000/studio."





"The query `*[type == 'project']` is applied to fetch and display only the project type documents in Sanity."



"The Cart page has been successfully integrated and is now visible on the website, displaying user-selected items."



Product	Price	Quantity	Total	Remove
 Burger	\$10.99	<input type="text" value="2"/>	\$21.98	×
 Fresh Lime	\$3.49	<input type="text" value="1"/>	\$3.49	×

"The 'Add to Cart' functionality is now active, enabling users to select and add products to their shopping cart."

