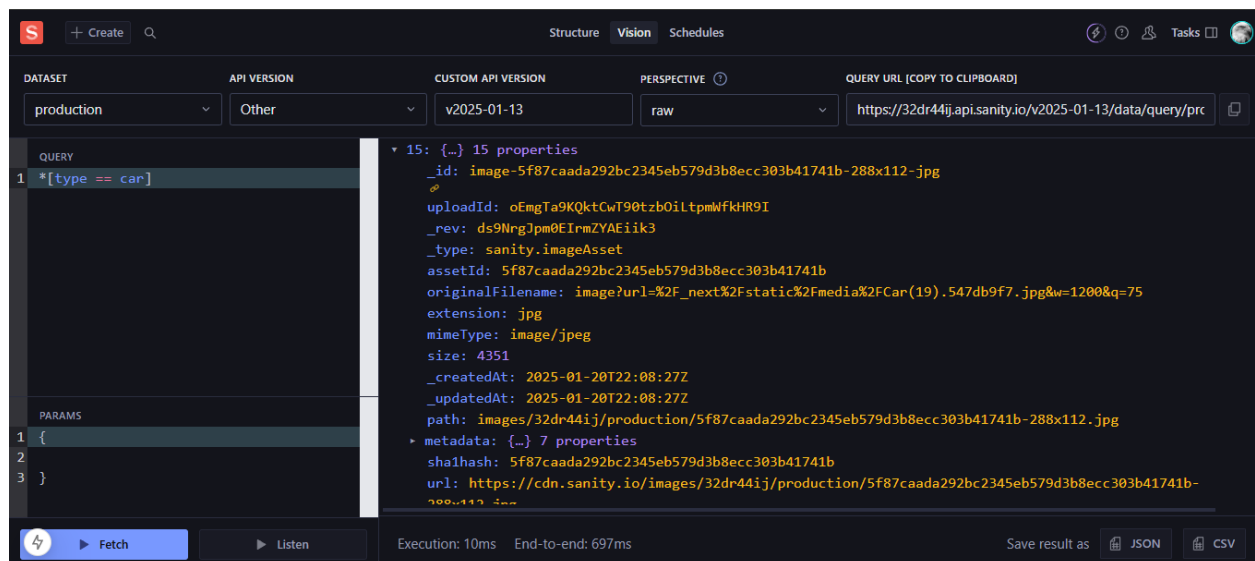


Day 3 - API Integration and Data Migration

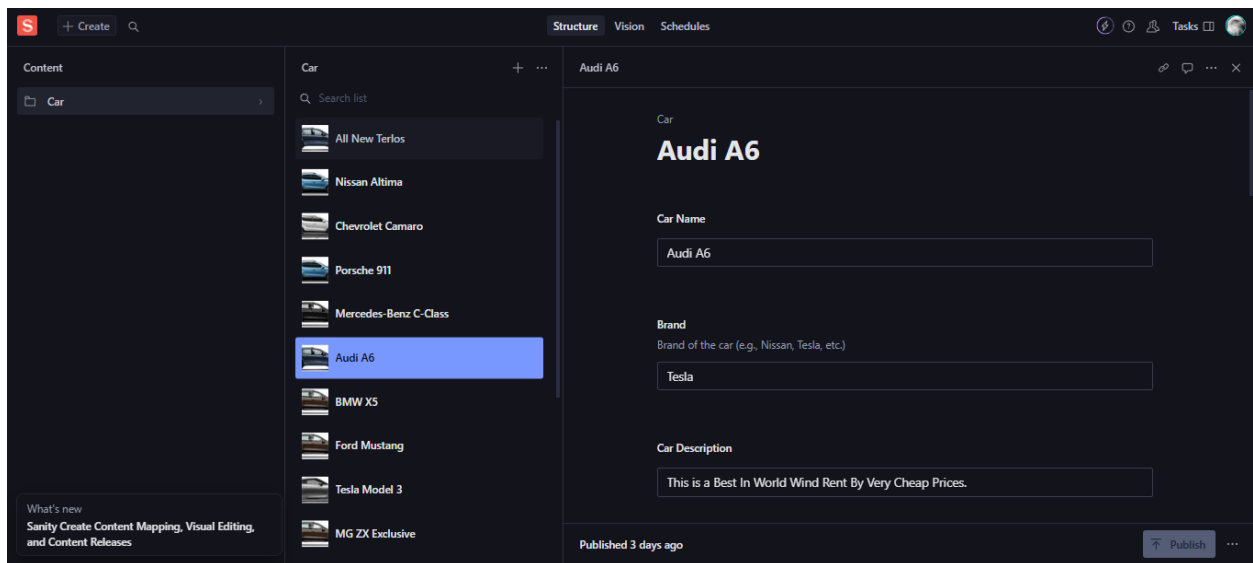
IMPLEMENT :

This activity focused on integrating APIs into a Next.js project and migrating data into Sanity CMS. The key objectives included validating schemas for compatibility, fetching and displaying data from provided APIs, and ensuring smooth data migration. The process also involved testing API calls, implementing error handling, and preparing a detailed report with screenshots and code snippets.

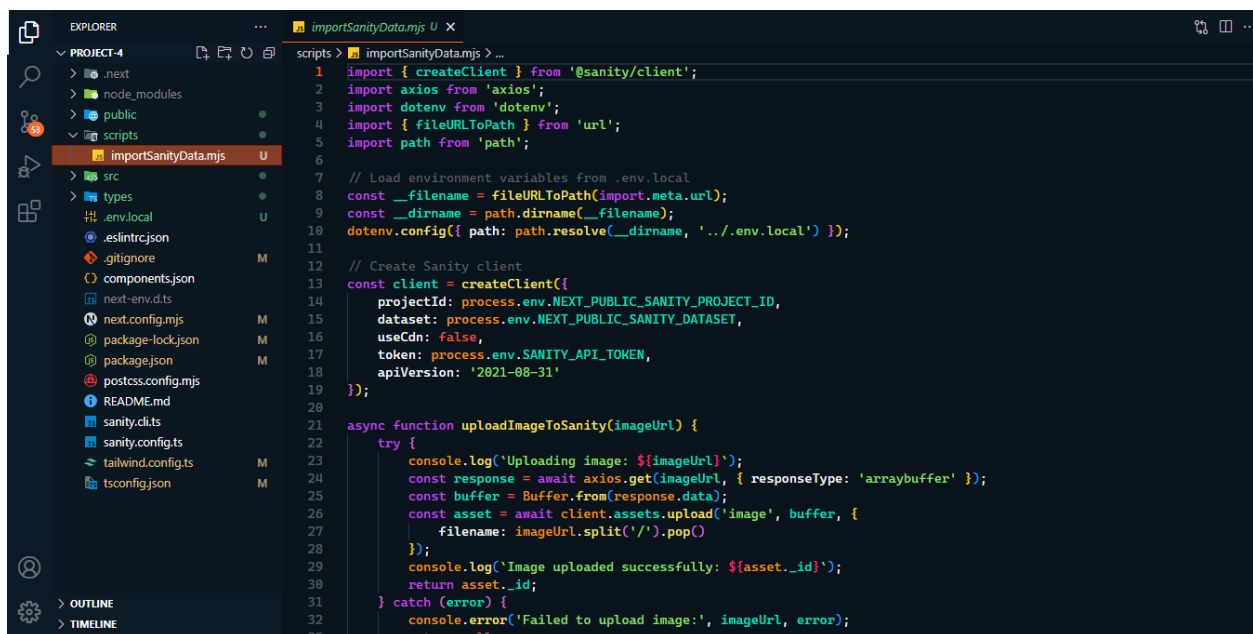
API Call Out Put



Sanity Data Import



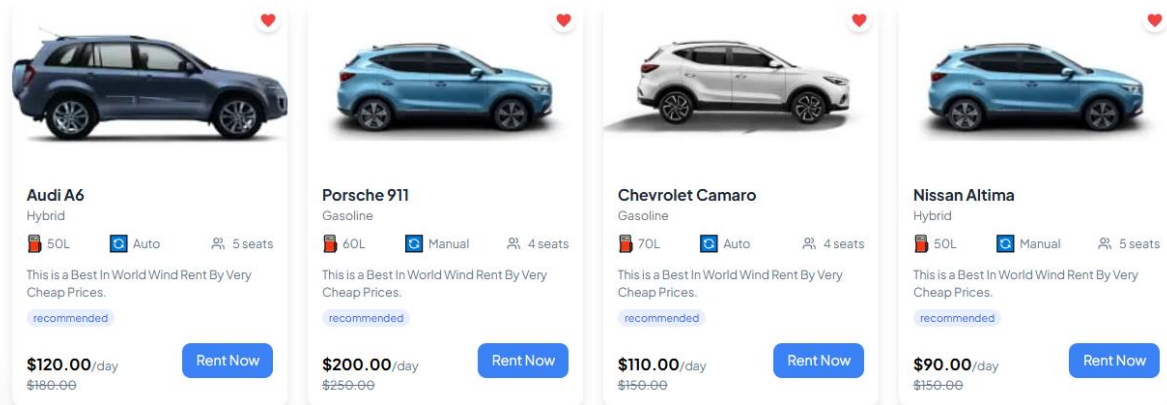
Import Sanity Data.mjs



Sanity Data Display

Sanity Car

[View All](#)



Sanity Data Display Render Code

```

src > app > display > car.tsx > CarDisplay > useEffect() callback
1 // Updated CarDisplay component (car.tsx)
2 "use client"; // Add this line to mark the component as a client component
3
4 import React, { useEffect, useState } from 'react';
5 import Image from 'next/image';
6 import { createClient } from '@sanity/client';
7 import { Users } from 'lucide-react';
8 import { urlFor } from '@sanity/lib/image';
9 import { useRouter } from 'next/navigation'; // Correct import for useRouter in App Router setup
10
11 // Sanity client setup
12 const client = createClient({
13   projectId: '32dr44ij', // Replace with your Sanity project ID
14   dataset: 'production',
15   apiVersion: "2023-01-01",
16   useCdn: true,
17 });
18
19 // Car interface for TypeScript
20 interface Car {
21   _id: string;
22   name: string;
23   brand: string;
24   description: string;
25   type: string;
26   fuelCapacity: string;
27   transmission: string;
28   seatingCapacity: string;
29   pricePerDay: string;
30   originalPrice: string;
31   tags: string[];
32   image?: {

```

Car Type Page Code

```
car.ts  U x
types > car.ts > Car > slug
1
2 export interface Car {
3   _id : string
4   name: string
5   brand: string
6   description: string
7   type: string
8   fuelCapacity: string
9   transmission: string
10  seatingCapacity: string
11  pricePerDay: string
12  originalPrice: string
13  tags: string[]
14  image: string
15  liked: boolean
16  rating: number
17  slug: {
18    current: string;
19  };
20 }
```

Car Queries Page Code

```
car.ts  U  queries.ts  U  X
src > sanity > lib > queries.ts > ...
1  import { groq } from 'next-sanity';
2
3  export const popularCars = groq`*[_type == "car"][4..7]`;
4  export const allcars = groq`*[_type == "car"]`;
```

Car Fields Page Code

```
car.ts  U  X
src > sanity > schemasTypes > car.ts > ...
1  export const car = {
2    name: 'car',
3    type: 'document',
4    title: 'Car',
5    fields: [
6
7      {
8        name: 'name',
9        type: 'string',
10       title: 'Car Name',
11     },
12     {
13       name: 'brand',
14       type: 'string',
15       title: 'Brand',
16       description: 'Brand of the car (e.g., Nissan, Tesla, etc.)',
17     },
18     {
19       name: 'description',
20       type: 'string',
21       title: 'Car Description',
22     },
23     {
24       name: 'type',
25       type: 'string',
26       title: 'Car Type',
27       description: 'Type of the car (e.g., Sport, Sedan, SUV, etc.)',
28     },
29     {
30       name: 'fuelCapacity',
31       type: 'string',
32       title: 'Fuel Capacity',
```

Env.Local Page Code

```
1  NEXT_PUBLIC_SANITY_PROJECT_ID="32dr4Uij"  
2  NEXT_PUBLIC_SANITY_DATASET="production"  
3  SANITY_API_TOKEN=skt2LBkKfPM7D3EQfbSc2EA1XZnvGEgrSTIbDHDz9HKzR5Q7YbJZf7tXOfJaE0p8datJepHS9c19ukHIeK0Jj0MpUJha0gaYaHB6sQu2mhMaN7CA48nIoJw67NCHabc4qY0XS  
4  
5
```

Json Page Code

```
5  "scripts": {  
6    "dev": "next dev",  
7    "build": "next build",  
8    "start": "next start",  
9    "lint": "next lint",  
10   "import-data": "node scripts/importSanityData.mjs"  
11  },
```