

```
1 export const car = {
2   name: "car",
3   type: "document",
4   title: "Car",
5   fields: [
6     {
7       name: "name",
8       type: "string",
9       title: "Car Name",
10    },
11    {
12      name: "brand",
13      type: "string",
14      title: "Brand",
15      description: "Brand of the car (e.g., Nissan, Tesla, etc.)",
16    },
17    {
18      name: "type",
19      type: "string",
20      title: "Car Type",
21      description: "Type of the car (e.g., Sport, Sedan, SUV, etc.)",
22    },
23    {
24      name: "fuelCapacity",
25      type: "string",
26      title: "Fuel Capacity",
27      description: "Fuel capacity or battery capacity (e.g., 90L, 100kWh)",
28    },
29    {
30      name: "transmission".
```

```
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: process.env.NEXT_PUBLIC_SANITY_PROJECT_ID,
15   dataset: process.env.NEXT_PUBLIC_SANITY_DATASET,
16   useCdn: false,
17   token: process.env.SANITY_API_TOKEN,
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() {
```