

## **QUESTION 3**

### **Building Joy's Café Search UI**

#### **Problem Statement:**

Joy is delighted with the simple search method developed in the previous section and now wishes to enhance it by creating a user interface connected to a real data source through a REST API.

#### **Requirements:**

1. Develop a single-page application using HTML and vanilla JavaScript. Avoid using any JavaScript frameworks or libraries.
2. A search box should be provided where Joy can search for cafes by name (case-insensitive). Search results should dynamically update with each keystroke, using the searched term as a substring.
3. Just below the search box, a table should display the cafe list.
4. When the search box is empty, the table should show all cafes. As the user enters text in the search box, it should display the filtered results.
5. On page load, fetch cafes and places from the given endpoints and use them to generate the cafe list.

#### **API Endpoints:**

- GET REQUEST: [Cafes Endpoint](#)
- GET REQUEST: [Places Endpoint](#)

#### **Note:**

Special credits will be given for:

- Clean and properly structured code.
- Optimized performance.
- Lesser runtime complexity.