**Q1: Can you create a simple HTML search input with a button to submit a keyword?**

****

<!DOCTYPE html>

<html>

<head>

<title>Search Page</title>

</head>>

<body>

<h1>Search Page</h1>

<form>

<input type="text" id="searchInput" name="searchInput">

<button type="submit" onclick="search()">Search</button>

</form>

<div id="searchResults"></div>

<script>

function search() {

var keyword = document.getElementById("searchInput").value;

fetch("https://www.example.com/search?q=" + keyword)

.then(response => response.text())

.then(data => {

document.getElementById("searchResults").innerHTML = data;

});

}

</script>

</body>

</html>

**Q2: Build an HTML search bar styled with CSS that filters a list of items in real time using JavaScript as the user types.**

****

<div class="search-bar">

<input type="text" id="search-input" placeholder="Search...">

<div class="search-results"></div>

</div>

```

CSS:

```css

.search-bar {

width: 300px;

height: 40px;

border: 1px solid #ccc;

box-shadow: 0px 0px 10px rgba(0,0,0,0.2);

}

.search-input {

width: 100%;

padding: 10px;

font-size: 16px;

border: none;

border-radius: 5px;

box-sizing: border-box;

}

.search-results {

width: 200px;

height: 300px;

margin-top: 20px;

padding: 10px;

border: 1px solid #ccc;

box-shadow: 0px 0px 10px rgba(0,0,0,0.2);

}

```

JavaScript (using jQuery for simplicity):

```javascript

$(document).ready(function() {

// Create a function to filter the results based on the search input

function filterResults(results) {

var searchInput = $("#search-input").val();

var filteredResults = [];

$.each(results, function(index, result) {

if ($(result).text().includes(searchInput)) {

filteredResults.push(result);

}

});

return filteredResults;

}

// Set up the search bar to filter the results in real time

$("#search-input").on("keyup", function() {

var searchResults = $( "#results" ).html();

var filteredResults = filterResults(searchResults);

$( ".search-results" ).html(filteredResults );

});

});

**Q3: Create a fully styled HTML, CSS, and JavaScript search interface that allows users to filter a dynamic list of items using keywords and highlights the matching results.**

****

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Search Interface</title>

<link rel="stylesheet" href="styles.css">

</head>

<body>

<div class="container">

<h1>Search Interface</h1>

<form>

<input type="text" id="search-query" placeholder="Search...">

<button type="submit">Search</button>

</form>

<ul id="results"></ul>

<script src="script.js"></script>

</div>

</body>

</html>

```

CSS:

```css

\* {

box-sizing: border-box;

}

body {

font-family: Arial, sans-serif;

margin: 0;

padding: 0;

}

.container {

display: flex;

flex-direction: column;

align-items: center;

gap: 20px;

}

h1 {

text-align: center;

}

form {

display: flex;

flex-direction: column;

align-items: center;

gap: 20px;

}

input[type="text"] {

width: 80%;

padding: 10px;

border: none;

border-radius: 5px;

box-shadow: 0 0 10px rgba(0, 0, 0, 0.2);

}

button {

width: 18%;

padding: 10px;

background-color: #4CAF50;

color: white;

border: none;

border-radius: 5px;

cursor: pointer;

}

#results {

list-style-type: none;

padding: 0;

margin: 20px 0;

}

#results li {

padding: 10px;

border-bottom: 1px solid #ccc;

}

#results li:last-child {

border-bottom: none;

}

```

JavaScript:

```javascript

const searchQuery = document.getElementById('search-query');

const results ul = document.getElementById('results');

function filterResults(query) {

const filteredItems = [];

const items = [...yourDynamicListOfItems];

for (const item of items) {

if (item.name.includes(query)) {

filteredItems.push(item);

}

}

return filteredItems;

}

function updateResults() {

const query = searchQuery.value.toLowerCase();

const filteredResults = filterResults(query);

results ul.innerHTML = '';

for (const item of filteredResults) {

const li = document.createElement('li');

li.textContent = item.name;

results ul.appendChild(li);

}

}

searchQuery.addEventListener('input', updateResults);