// ====================================

// 1. Employee Data (Simulated Database)

// ====================================

const employees = [

{

id: 1,

name: "Alice Johnson",

title: "Senior Software Engineer",

department: "Technology",

contact: "alice.j@corp.com",

photo: "https://i.pravatar.cc/150?img=1" // Placeholder Image

},

{

id: 2,

name: "Bob Williams",

title: "Marketing Specialist",

department: "Marketing",

contact: "bob.w@corp.com",

photo: "https://i.pravatar.cc/150?img=2"

},

{

id: 3,

name: "Charlie Brown",

title: "HR Manager",

department: "Human Resources",

contact: "charlie.b@corp.com",

photo: "https://i.pravatar.cc/150?img=3"

},

{

id: 4,

name: "Diana Prince",

title: "Financial Analyst",

department: "Finance",

contact: "diana.p@corp.com",

photo: "https://i.pravatar.cc/150?img=4"

},

{

id: 5,

name: "Edward Norton",

title: "Product Designer",

department: "Design",

contact: "edward.n@corp.com",

photo: "https://i.pravatar.cc/150?img=5"

},

{

id: 6,

name: "Fiona Glenn",

title: "Data Scientist",

department: "Technology",

contact: "fiona.g@corp.com",

photo: "https://i.pravatar.cc/150?img=6"

}

// Add more employee objects here

];

// ====================================

// 2. DOM Elements and Event Listener

// ====================================

const employeeList = document.getElementById('employee-list');

const searchInput = document.getElementById('searchInput');

const noResultsMessage = document.getElementById('no-results');

// Attach the filtering function to the search input's 'keyup' event

searchInput.addEventListener('keyup', filterEmployees);

// ====================================

// 3. Rendering Function

// ====================================

/\*\*

\* Renders the provided array of employee data to the DOM.

\* @param {Array} data - The array of employee objects to display.

\*/

function renderEmployees(data) {

// Clear the current list content

employeeList.innerHTML = '';

if (data.length === 0) {

// If no employees are found, show the 'no results' message

noResultsMessage.style.display = 'block';

return;

}

// Hide the 'no results' message if there is data

noResultsMessage.style.display = 'none';

// Loop through the data and create an HTML card for each employee

data.forEach(employee => {

const cardHTML = `

<div class="employee-card" data-id="${employee.id}">

<img src="${employee.photo}" alt="${employee.name}" class="employee-photo">

<h2 class="employee-name">${employee.name}</h2>

<p class="employee-title">${employee.title}</p>

<p class="employee-department">Dept: ${employee.department}</p>

<p class="employee-contact">Email: ${employee.contact}</p>

</div>

`;

// Append the card HTML to the employee list container

employeeList.innerHTML += cardHTML;

});

}

// ====================================

// 4. Search/Filter Logic

// ====================================

/\*\*

\* Filters the main employee array based on the current search input value

\* and re-renders the list.

\*/

function filterEmployees() {

// 1. Get the search term and convert it to lowercase for case-insensitive search

const searchTerm = searchInput.value.toLowerCase();

// 2. Filter the main 'employees' array

const filteredList = employees.filter(employee => {

// Check if the search term is found in the employee's name, title, OR department

const nameMatch = employee.name.toLowerCase().includes(searchTerm);

const titleMatch = employee.title.toLowerCase().includes(searchTerm);

const deptMatch = employee.department.toLowerCase().includes(searchTerm);

// An employee matches if the search term is found in any of these fields

return nameMatch || titleMatch || deptMatch;

});

// 3. Render the filtered list

renderEmployees(filteredList);

}

// ====================================

// 5. Initial Load

// ====================================

// Render the full list when the page first loads

document.addEventListener('DOMContentLoaded', () => {

renderEmployees(employees);

});