**Exercise 4: Employee Management System**

**Scenario:**

**You are developing an employee management system for a company. Efficiently managing employee records is crucial.**

**Steps:**

1. **Understand Array Representation:**

* **Explain how arrays are represented in memory and their advantages.**

**Array Representation in Memory**

1. *Contiguous Memory Allocation:* Arrays are stored in contiguous memory locations, which means all elements are placed next to each other in memory.
2. *Fixed Size:* The size of an array is fixed at the time of its creation and cannot be changed dynamically.
3. *Index-Based Access:* Each element in an array can be accessed using its index, allowing for O(1) time complexity for element access.

**Advantages of Arrays**

1. *Fast Access:* O(1) time complexity for accessing elements by index.
2. *Memory Efficiency:* Arrays have low memory overhead compared to other data structures like linked lists.
3. *Simplicity:* Arrays are simple to use and understand, making them a good choice for basic data storage needs.
4. **Setup:**
   * **Create a class Employee with attributes like employeeId, name, position, and salary.**
5. **Implementation:**
   * **Use an array to store employee records.**
   * **Implement methods to add, search, traverse, and delete employees in the array.**
6. **Analysis:**

* **Analyze the time complexity of each operation (add, search, traverse, delete).**
* Add Operation: O (1) if array is not full.
* Search Operation: O(n) since it may need to check each element in the worst case.
* Traverse Operation: O(n) as it involves visiting each element in the array.
* Delete Operation: O(n) since it may need to shift elements to fill the gap created by the deletion.
* **Discuss the limitations of arrays and when to use them.**

1. Arrays have a fixed size, which reduces their flexibility. If array becomes full, no new elements can be added unless a new array is created of size bigger than the previous array.
2. For inserting and deleting elements in the middle it requires O(n) time complexity which makes it inefficient.
3. If array contains less elements than its max size then there is wastage of space.

Arrays can be used when: -

1. When the number of elements is known in advance.
2. When fast access of individual elements is required.