**Exercise 4: Employee Management System**

**Array Representation**:

* **Memory Layout**: In Java, arrays are contiguous blocks of memory where each element is of the same type. The elements are indexed starting from 0. For instance, if an array of integers is created, all integers are stored in adjacent memory locations.
* **Advantages**:
  + **Fast Access**: Arrays allow direct access to elements using indices, making retrieval operations very fast (O(1)O(1)O(1) time complexity).
  + **Simplicity**: They are simple to implement and use for scenarios where the size is known and fixed.
  + **Low Overhead**: Arrays have less memory overhead compared to other data structures like linked lists.

**Implementation**:

Please refer the code.

**Analysis**

**Time Complexity**:

1. **Add Operation**:
   * **Best Case**: O(1)O(1)O(1) - When there is space in the array.
   * **Worst Case**: O(1)O(1)O(1) - Adding an element is constant time if space is available.
2. **Search Operation**:
   * **Best Case**: O(1)O(1)O(1) - When the employee is at the beginning of the array.
   * **Worst Case**: O(n)O(n)O(n) - When the employee is at the end or not present.
3. **Traverse Operation**:
   * **Time Complexity**: O(n)O(n)O(n) - Requires visiting every element.
4. **Delete Operation**:
   * **Best Case**: O(n)O(n)O(n) - If the employee is at the end of the array.
   * **Worst Case**: O(n)O(n)O(n) - Shifting elements after deletion.

**Limitations of Arrays**:

1. **Fixed Size**: Arrays have a fixed size once initialized. Adding more elements requires creating a new array and copying existing elements, which can be inefficient.
2. **Insertion and Deletion**: Adding or removing elements requires shifting elements, which can be slow for large arrays.
3. **Memory Usage**: Arrays allocate memory for a fixed size regardless of the number of elements actually stored, leading to possible wasted space.

**When to Use Arrays**:

* **When Size is Known**: Arrays are suitable when the maximum size is known and does not change frequently.
* **For Simple Scenarios**: Arrays are appropriate for simple use cases where operations are straightforward and performance requirements are modest