**Exercise 5: Task Management System**

**Scenario:**

You are developing a task management system where tasks need to be added, deleted, and traversed efficiently.

**Steps:**

1. **Understand Linked Lists:**
   * Explain the different types of linked lists (Singly Linked List, Doubly Linked List).
2. **Setup:**
   * Create a class Task with attributes like taskId, taskName, and status.
3. **Implementation:**
   * Implement a singly linked list to manage tasks.
   * Implement methods to add, search, traverse, and delete tasks in the linked list.
4. **Analysis:**
   * Analyze the time complexity of each operation.
   * Discuss the advantages of linked lists over arrays for dynamic data.

**ANSWER:**

**Explain the different types of linked lists (Singly Linked List, Doubly Linked List).**

**Singly Linked List:**

* Each node in a singly linked list contains a reference to the next node in the sequence.
* It allows traversal in one direction: from the head to the last node.

**Doubly Linked List:**

* Each node contains references to both the next and the previous nodes.
* It allows traversal in both directions: forward and backward.

**Analyze the time complexity of each operation.**

| **Cases** | **Add employee** | **Search Employee** | **Traverse Employee** | **Delete Employee** |
| --- | --- | --- | --- | --- |
|  | O(n) | O(n) | O(n) | O(n) |

**Discuss the advantages of linked lists over arrays for dynamic data.**

Linked lists can easily grow and shrink in size by simply adjusting pointers, whereas arrays require resizing and copying elements.

Inserting or deleting elements in a linked list is more efficient (O(1) for inserting at the head or deleting from the head) compared to arrays (O(n) for inserting or deleting at any position).