**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).

Solution:

1. Singly Linked List:
   * Each node contains a data element and a reference to the next node.
   * Allows efficient insertion and deletion but only traverses in one direction.
2. Doubly Linked List:
   * Each node contains a data element, a reference to the next node, and a reference to the previous node.
   * Allows traversal in both directions but uses more memory.
3. **Setup:**
   * Create a class **Task** with attributes like **taskId**, **taskName**, and **status**.
4. **Implementation:**
   * Implement a singly linked list to manage tasks.
   * Implement methods to **add**, **search**, **traverse**, and **delete** tasks in the linked list.

Solution:

Setup and Implementation part is in attached java file.

1. **Analysis:**
   * Analyze the time complexity of each operation.

Solution:

Time Complexity:

* Add Operation: O(n) in the worst case, because you might need to traverse the list to add at the end.
* Search Operation: O(n) because you may need to traverse the entire list to find the task.
* Traverse Operation: O(n) because you must visit each node once.
* Delete Operation: O(n) because you may need to traverse the list to find the node to delete.
  + Discuss the advantages of linked lists over arrays for dynamic data.

Solution:

Advantages of Linked Lists:

* Dynamic Size: Can grow and shrink dynamically without needing to allocate or deallocate large blocks of memory.
* Efficient Insertions/Deletions: Insertions and deletions are more efficient than arrays when the location is known because they don't require shifting elements.