## **Exercise 4: Task Management System**

A Linked List is a linear data structure where each element (node) contains data and a reference (link) to the next node in the sequence.

## **Types of Linked Lists**

### **1. Singly Linked List**

* In this type, **each node points to the next node only**.
* It moves in **one direction**, from the head (start) to the end.
* The last node points to null.
* **Structure:**

data → next

**Example:** 10 → 20 → 30 → null

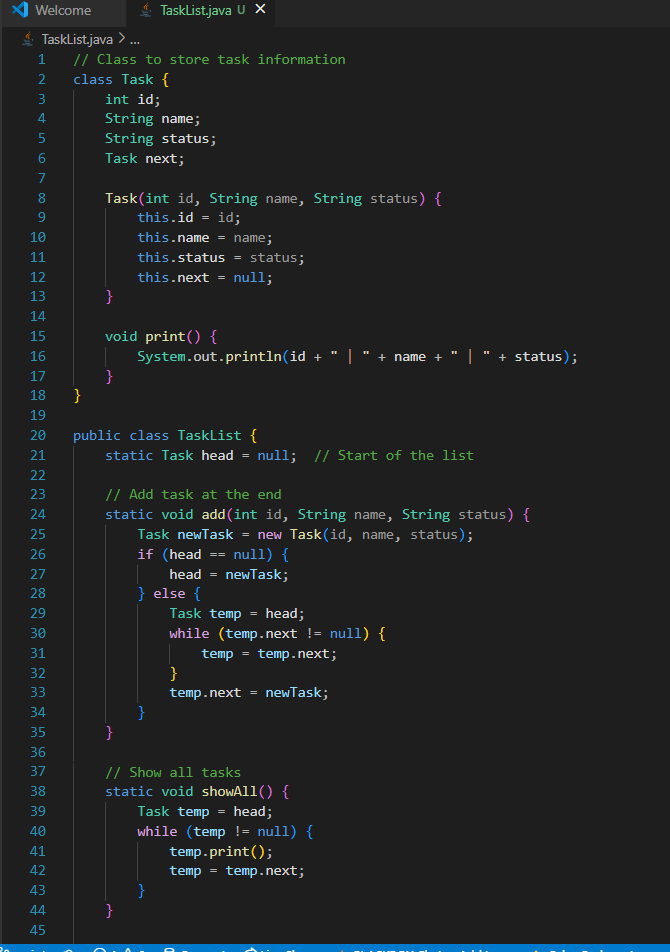
**➝ Operations like adding, deleting, and traversing are done in one forward direction.**

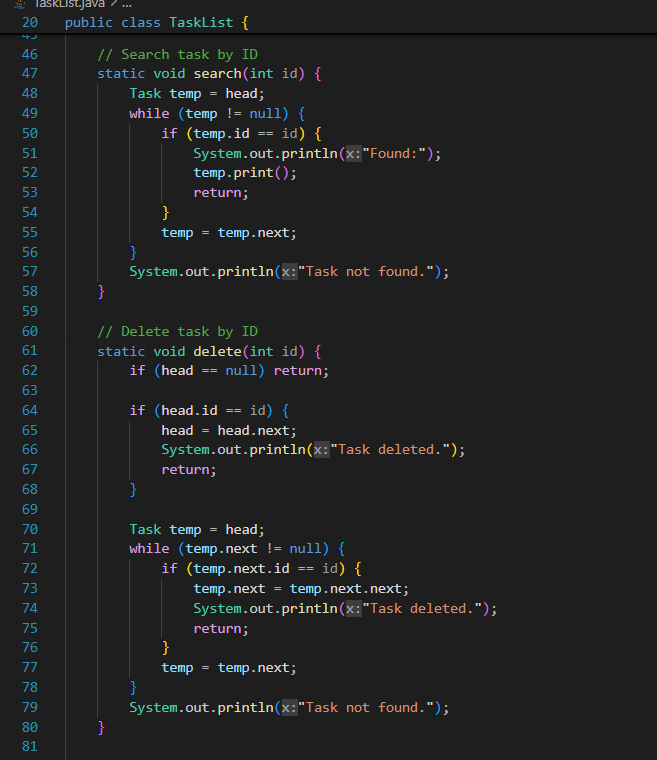
### **2.Doubly Linked List**

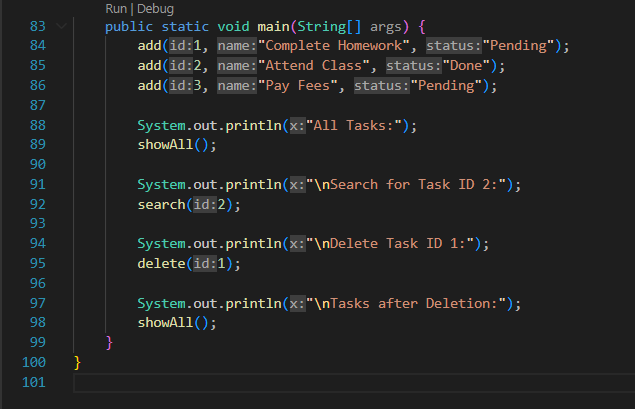
* Here, **each node points to both the next and the previous node**.
* It moves in **both directions**: forward and backward.
* The first node’s previous is null, and the last node’s next is null.
* **Structure:**

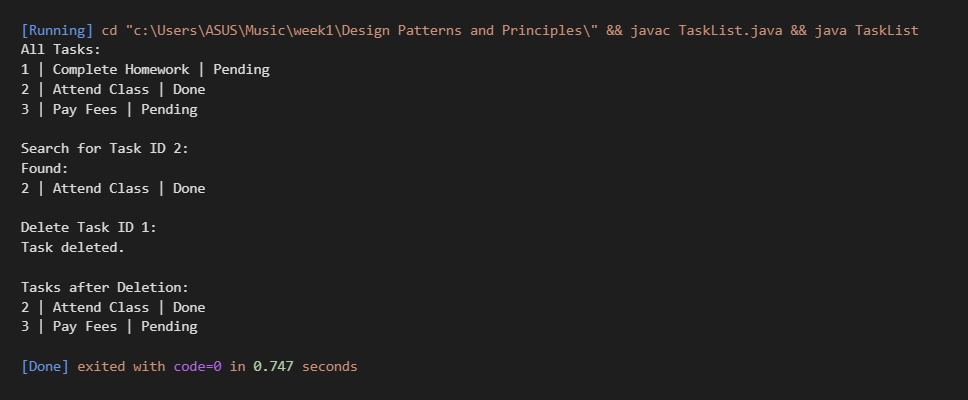
prev ← data → next

✅ **Example:** null ← 10 ↔ 20 ↔ 30 → null









## **Time Complexity in Singly Linked List**

|  |  |  |
| --- | --- | --- |
| **Operation** | **Time Complexity** | **Reason** |
| **Add (at end)** | O(n) | Need to traverse to the last node. |
| **Add (at start)** | O(1) | Directly link the new node to head. |
| **Search** | O(n) | Check each node one by one. |
| **Traverse (Display all)** | O(n) | Visit every node once. |
| **Delete (by value or ID)** | O(n) | Need to find the node first, then delete. |

*Most operations in a linked list take O(n) time except inserting at the start, which is O(1).*

## **Advantages of Linked Lists over Arrays for Dynamic Data**

**1.Dynamic Size:**

Linked lists can easily grow or shrink at runtime, while arrays have a fixed size.

2.**Easy Insertion/Deletion:**

In linked lists, inserting or deleting elements is easier as you only need to change the links, without shifting other elements like in arrays.

**3.Efficient Memory Usage:**

Linked lists use memory only when needed. Arrays may waste memory if declared larger than required.

4. **No Wastage of Space:**

You don’t need to pre-define the size, avoiding unused or excess space.

5. **No Continuous Memory Required:**

Linked lists can be scattered in memory, unlike arrays which need a continuous block.