Leetcode Problem 2. (Easy)

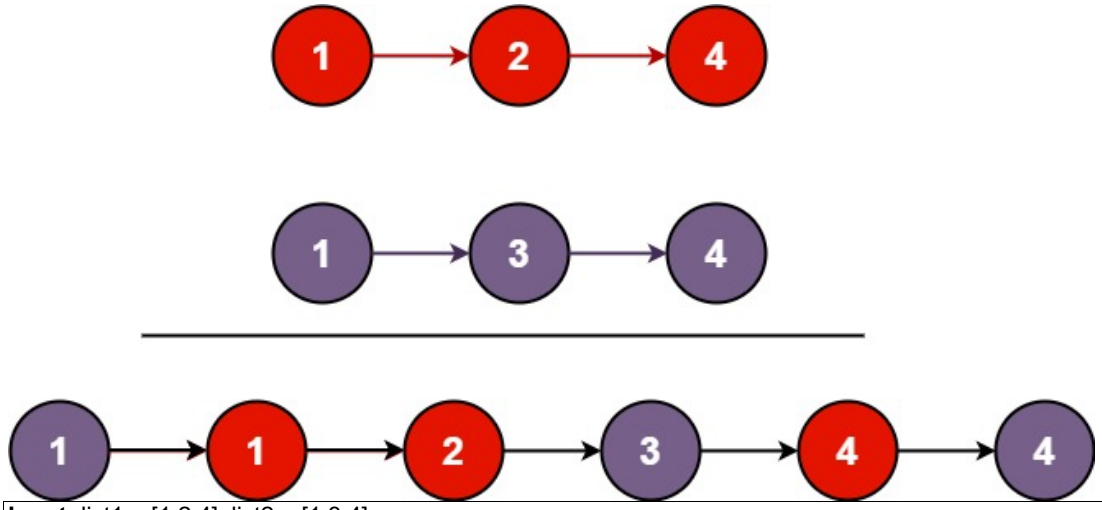
Merge Two Sorted Lists

You are given the heads of two sorted linked lists list1 and list2.

Merge the two lists in a one **sorted** list. The list should be made by splicing together the nodes of the first two lists.

Return *the head of the merged linked list*.

Example 1:



Input: list1 = [1,2,4], list2 = [1,3,4]
Output: [1,1,2,3,4,4]

Example 2:

Input: list1 = [], list2 = []

Output: []

Example 3:

Input: list1 = [], list2 = [0]
Output: [0]

Constraints:

- The number of nodes in both lists is in the range [0, 50].
- -100 <= Node.val <= 100
- Both list1 and list2 are sorted in **non-decreasing** order.

Link : https://leetcode.com/problems/merge-two-sorted-lists/

```java
public class Solution {
    public ListNode mergeTwoLists(ListNode list1, ListNode list2) {
        ListNode mergedList = new ListNode(-1);
        ListNode current = mergedList;

        while (list1 != null && list2 != null)
        {
            if (list1.val <= list2.val) {
                current.next = list1;
                list1 = list1.next;
            } else {
                current.next = list2;
                list2 = list2.next;
            }
            current = current.next;
        }
        if (list1 != null) {
            current.next = list1;
        } else {
            current.next = list2;
        }

        return mergedList.next;
    }
}
```