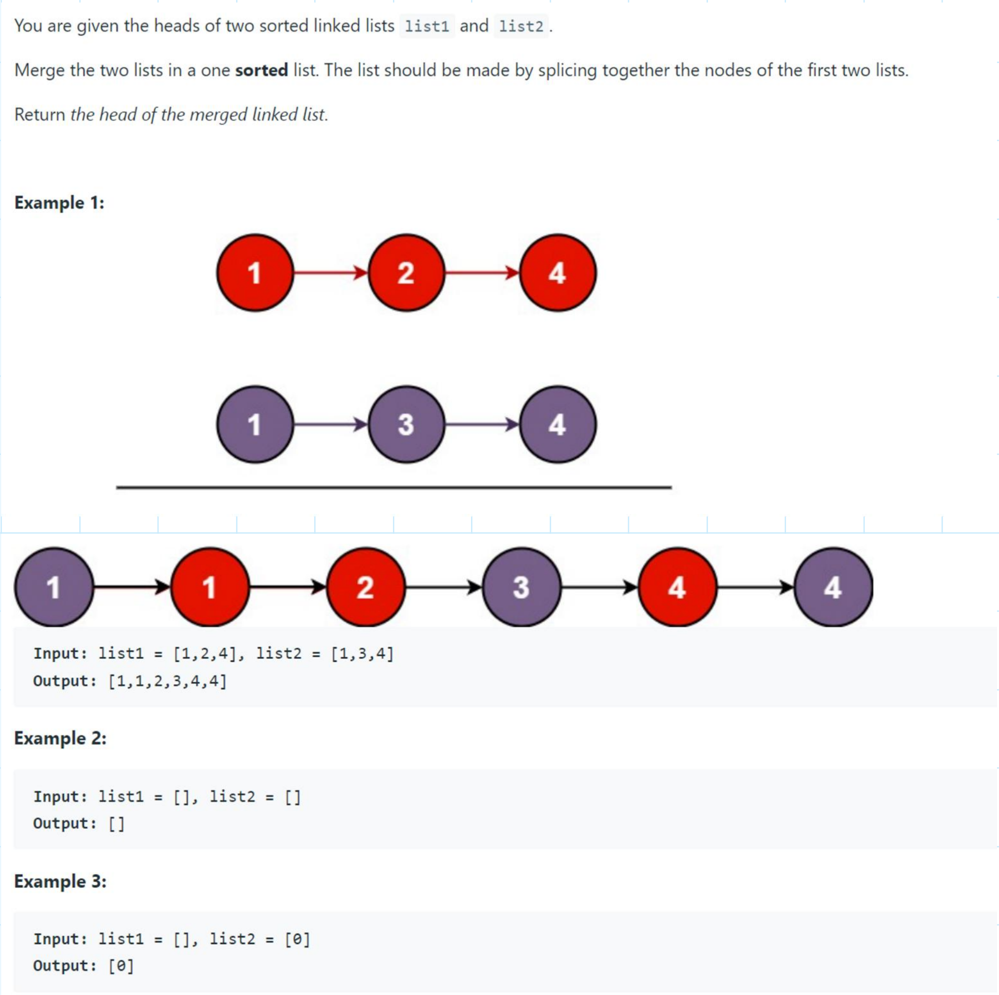
21. Merge Two Sorted Lists

17 December 2021 08:39 PM



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]
```

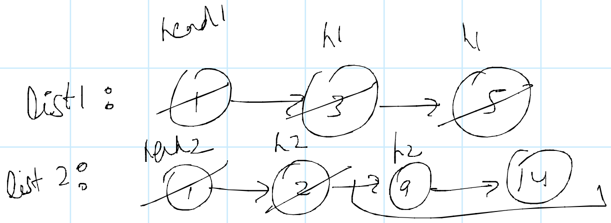
Example 2:

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

Example 3:

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

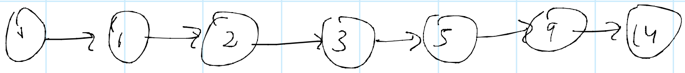
+ Both the lists are arranged in non-decreasing order.

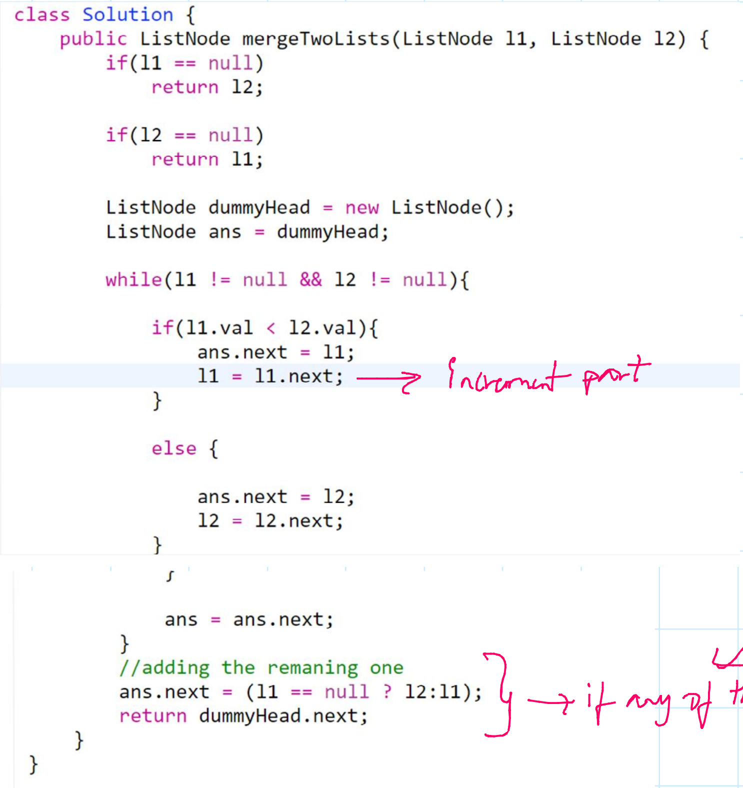


+ Compare both the list and move the head.

+ if any of the list become empty add the other list value to the ans.

ans:





```java
class Solution {
    public ListNode mergeTwoLists(ListNode l1, ListNode l2) {
        if(l1 == null)
            return l2;

        if(l2 == null)
            return l1;

        ListNode dummyHead = new ListNode();
        ListNode ans = dummyHead;

        while(l1 != null && l2 != null){

            if(l1.val < l2.val){
                ans.next = l1;
                l1 = l1.next;
            }

            else {

                ans.next = l2;
                l2 = l2.next;
            }

            ans = ans.next;
        }
        //adding the remaning one
        ans.next = (l1 == null ? l2:l1);
        return dummyHead.next;
    }
}
```

→ Increment part

} → if any of the list is empty