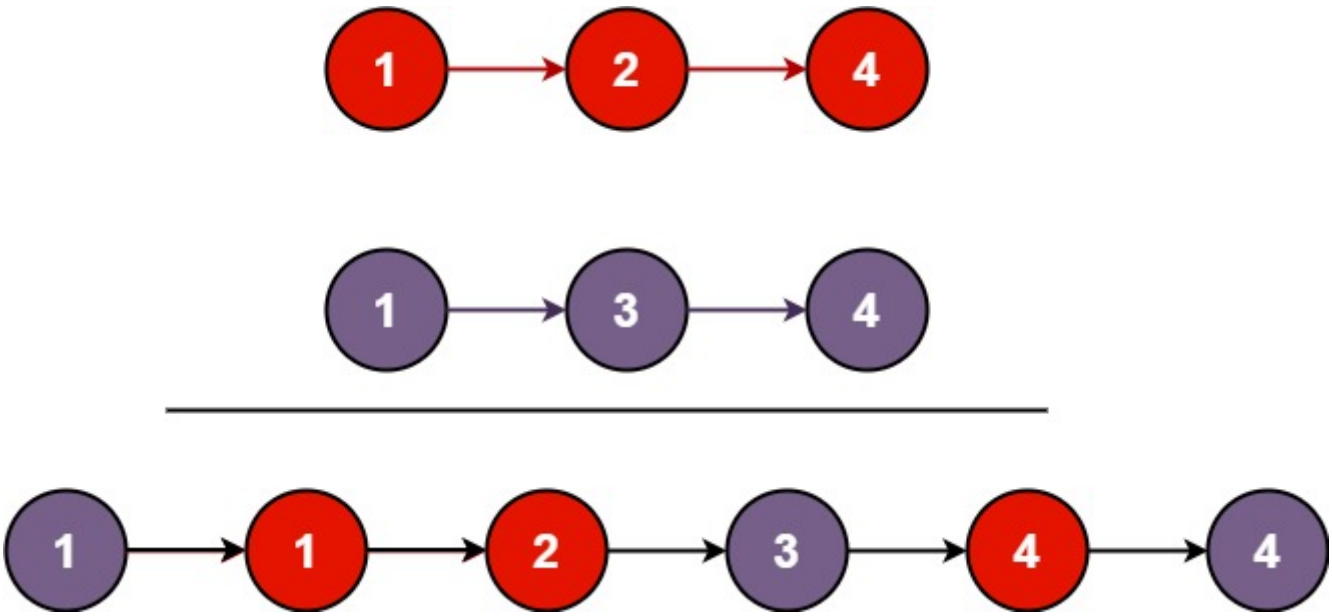
21. Merge Two Sorted Lists

Merge two sorted linked lists and return it as a **sorted** list. The list should be made by splicing together the nodes of the first two lists.

Example 1:




```
Input: l1 = [1,2,4], l2 = [1,3,4]
Output: [1,1,2,3,4,4]
```

Example 2:

```
Input: l1 = [], l2 = []
Output: []
```

Example 3:

```
Input: l1 = [], l2 = [0]
Output: [0]
```

Constraints:

- The number of nodes in both lists is in the range `[0, 50]`.
- `-100 <= Node.val <= 100`
- Both `l1` and `l2` are sorted in **non-decreasing** order.

```
# Definition for singly-linked list.
# class ListNode:
#     def __init__(self, val=0, next=None):
```

```
#         self.val = val
#         self.next = next
class Solution:
    def mergeTwoLists(self, l1: Optional[ListNode], l2: Optional[ListNode])
-> Optional[ListNode]:
        if l1 is None or l2 is None:
            return l1 if l2 is None else l2
        dummy = ListNode(0)
        prev = dummy


        c1 = l1
        c2 = l2

        while c1!=None and c2!=None:
            if c1.val>c2.val:
                prev.next = c2
                c2 = c2.next
            else:
                prev.next = c1
                c1 = c1.next
            prev = prev.next

        if c2 is None:
            prev.next = c1
        else:
            prev.next = c2
        return dummy.next
```

```
# Definition for singly-linked list.
# class ListNode:
#     def __init__(self, val=0, next=None):
#         self.val = val
#         self.next = next
class Solution:
    def mergeTwoLists(self, l1: ListNode, l2: ListNode) -> ListNode:
        dummy = ListNode(0)
        temp = dummy

        while l1 and l2:
            if l1.val<=l2.val:
                ListNode(l1.val)
                temp.next = ListNode(l1.val)
```

```
                temp = temp.next
                l1 = l1.next
            else:
                ListNode(l2.val)
                temp.next = ListNode(l2.val)
                temp = temp.next
                l2 = l2.next
        while l1!=None:
            temp.next = ListNode(l1.val)
            temp = temp.next
            l1 = l1.next
        while l2!=None:
            temp.next = ListNode(l2.val)
            temp = temp.next
            l2 = l2.next
        return dummy.next
```