Practice　🕐 **Compete**　💼 **Jobs**　🎖 **Rank**　🏆 **Leaderboard**

🔍　💬　📢　hin

Points: 373.91　Rank: 16540

Merge two sorted linked lists 🔖

by harsha_s

Problem	Submissions	Leaderboard	Discussions	Editorial

This challenge is part of a tutorial track by MyCodeSchool

You're given the pointer to the head nodes of two sorted linked lists. The data in both lists will be sorted in ascending order. Change the `next` pointers to obtain a single, merged linked list which also has data in ascending order. Either head pointer given may be null meaning that the corresponding list is empty.

Input Format

You have to complete the `Node* MergeLists(Node* headA, Node* headB)` method which takes two arguments - the heads of the two sorted linked lists to merge. You should NOT read any input from stdin/console.

Output Format

Change the `next` pointer of individual nodes so that nodes from both lists are merged into a single list. Then `return` the head of this merged list. Do NOT print anything to stdout/console.

Sample Input

```
1 -> 3 -> 5 -> 6 -> NULL
2 -> 4 -> 7 -> NULL

15 -> NULL
12 -> NULL

NULL
1 -> 2 -> NULL
```
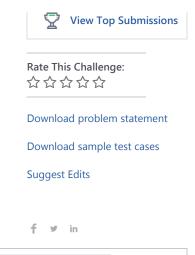
Sample Output

```
1 -> 2 -> 3 -> 4 -> 5 -> 6 -> 7 -> NULL
12 -> 15 -> NULL
1 -> 2 -> NULL
```

Explanation

1. We merge elements in both list in sorted order and output.

Easy　Submitted 54145 times
Max Score **5**

Need Help?

💬　**View Discussions**

💬　**View Editorial Solution**

Current Buffer (saved locally, editable) ⑂ 🕘   Python 3 ⌄

```python
"""
 Merge two linked lists
 head could be None as well for empty list
 Node is defined as

 class Node(object):

    def __init__(self, data=None, next_node=None):
        self.data = data
        self.next = next_node

 return back the head of the linked list in the below method.
"""

def MergeLists(headA, headB):
    if headA==None:
        return headB
    elif headB==None:
        return headA
    else:
        if headA.data>=headB.data:
            nhead=headB
            headB=headB.next
        else:
            nhead=headA
            headA=headA.next
        temp=nhead
        while(headA!=None and headB!=None):
            if headA.data>=headB.data:
                temp.next=headB
                headB=headB.next
            else:
                temp.next=headA
                headA=headA.next
            temp=temp.next
        while(headA):
            temp.next=headA
            headA=headA.next
        while(headB):
            temp.next=headB
            headB=headB.next
        return nhead
```

Line: 26 Col: 24

⬆ Upload Code as File   ☐ Test against custom input

Run Code   Submit Code

Testcase 0 ✔

Congratulations, you passed the sample test case.

Click the **Submit Code** button to run your code against all the test cases.

Input (stdin)

```
3
4
1 3 5 6
3
2 4 7
1
15
1
12
0
2
1 2
```

Your Output (stdout)

```
1 2 3 4 5 6 7
12 15
1 2
```

Expected Output

```
1 2 3 4 5 6 7
12 15
1 2
```