

Reverse a doubly linked list ■



This challenge is part of a tutorial track by MyCodeSchool

You're given the pointer to the head node of a doubly linked list. Reverse the order of the nodes in the list. The head node might be NULL to indicate that the list is empty.

Input Format

hy harsha_s

You have to complete the Node* Reverse(Node* head) method which takes one argument - the head of the doubly linked list. You should NOT read any input from stdin/console.

Output Format

Change the next and prev pointers of all the nodes so that the direction of the list is reversed. Then return the head node of the reversed list. Do NOT print anything to stdout/console.

Sample Input

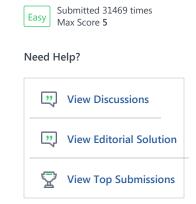
NULL

NULL <-- 2 <--> 4 <--> 6 --> NULL

Sample Output

Explanation

- 1. Empty list, so nothing to do.
- 2. 2,4,6 become 6,4,2 o reversing in the given doubly linked list.



```
Rate This Challenge:
☆☆☆☆☆

Download problem statement

Download sample test cases

Suggest Edits
```

y

```
Python 3
 Current Buffer (saved locally, editable) &
 1 🔻 """
 2
     Reverse a doubly linked list
 3
     head could be None as well for empty list
 4
     Node is defined as
 5
     class Node(object):
 6 ₹
 7
       def __init__(self, data=None, next_node=None, prev_node = None):
 8 ▼
 9
           self.data = data
10
           self.next = next_node
11
           self.prev = prev_node
12
13
     return the head node of the updated list
14
15 ▼ def Reverse(head):
        temp=head
16
17 ▼
        while(temp):
18
            head=temp
19
            head.next,head.prev=head.prev,head.next
20
            temp=temp.prev
21
        return head
22
23
24
25
26
27
28
29
30
31
                                                                                                                  Line: 18 Col: 18
```

1 Upload Code as File

Run Code

Submit Code

