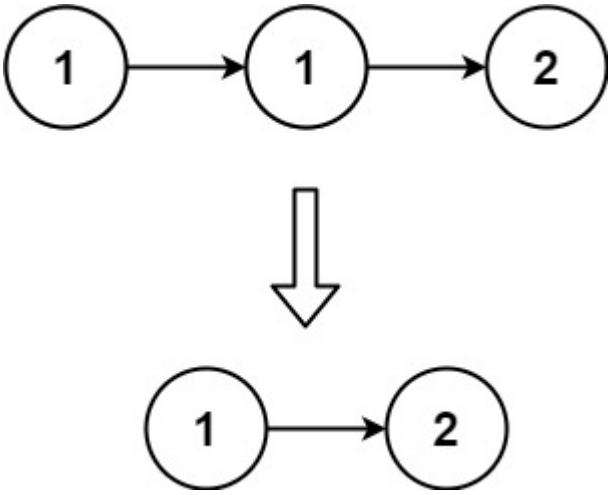
83. Remove Duplicates from Sorted List

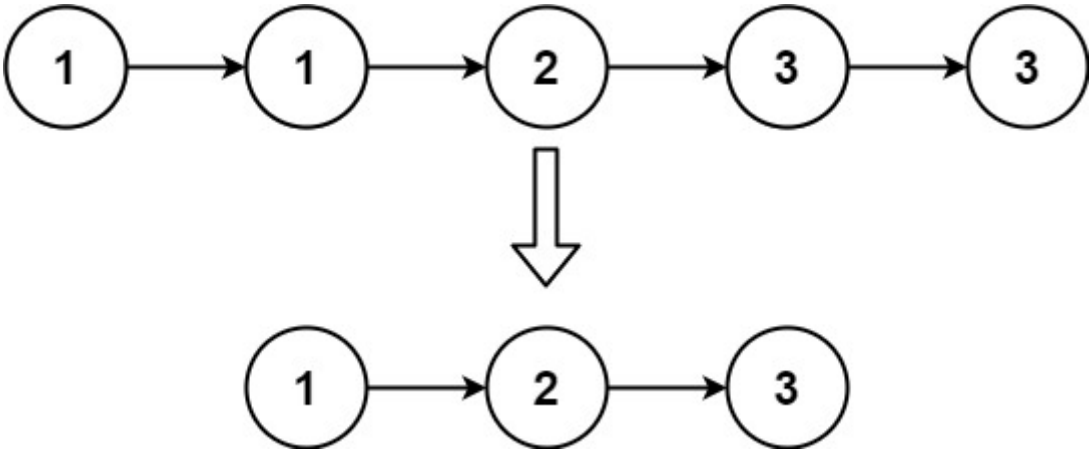
Given the `head` of a sorted linked list, *delete all duplicates such that each element appears only once*. Return *the linked list **sorted** as well*.

Example 1:




```
Input: head = [1,1,2]
Output: [1,2]
```

Example 2:




```
Input: head = [1,1,2,3,3]
Output: [1,2,3]
```

Constraints:

- The number of nodes in the list is in the range `[0, 300]`.
- `-100 <= Node.val <= 100`
- The list is guaranteed to be **sorted** in ascending order.

```python
def deleteDuplicates(self, head: Optional[ListNode]) -> Optional[ListNode]:
        if head is None or head.next is None:
            return head
        curr= head
        dummyHead = ListNode(-1)
        prev =dummyHead
        prevVal = -101

        while curr!=None:
            if prevVal!=curr.val:
                prevVal = curr.val
                prev.next = curr
                prev = curr
            curr = curr.next
        if prev.next!=None and prev.next.val==prev.val:
            prev.next = None
        return dummyHead.next
```