

Remove Duplicates from Sorted List

Given a sorted linked list, delete all duplicates such that each element appear only *once*.

For example,

Given `1->1->2`, return `1->2`.

Given `1->1->2->3->3`, return `1->2->3`.

Solution 1

This solution is inspired by renzid <https://leetcode.com/discuss/33043/3-line-recursive-solution>

```
public ListNode deleteDuplicates(ListNode head) {  
    if(head == null || head.next == null) return head;  
    head.next = deleteDuplicates(head.next);  
    return head.val == head.next.val ? head.next : head;  
}
```

Enjoy!

written by [wen587sort](#) original link [here](#)

Solution 2

```
public class Solution {  
    public ListNode deleteDuplicates(ListNode head) {  
        ListNode list = head;  
  
        while(list != null) {  
            if (list.next == null) {  
                break;  
            }  
            if (list.val == list.next.val) {  
                list.next = list.next.next;  
            } else {  
                list = list.next;  
            }  
        }  
  
        return head;  
    }  
}
```

written by [aleksey.danilevsky.5](#) original link [here](#)

Solution 3

I noticed that the solutions posted here are too long and complicated. They use unnecessary variables and/or checks etc. The solution can be much more concise. Here is my solution:

```
class Solution {
public:
    ListNode *deleteDuplicates(ListNode *head) {
        ListNode* cur = head;
        while (cur) {
            while (cur->next && cur->val == cur->next->val)
                cur->next = cur->next->next;
            cur = cur->next;
        }
        return head;
    }
};
```

Note about freeing memory. We need to free memory when we delete a node. But don't use `delete node;` construct on an interview without discussing it with the interviewer. A list node can be allocated in many different ways and we can use `delete node;` only if we are sure that the nodes were allocated with `new TreeNode(...);`.

written by [alexander4](#) original link [here](#)

From [LeetCoder](#).