

# 083. Remove Duplicates from Sorted List

## 083 Remove Duplicates from Sorted List

- Linked List

### Description

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`.

### 1. Thought line

### 2. Linked List

```
1 /**
2  * Definition for singly-linked list.
3  * struct ListNode {
4  *     int val;
5  *     ListNode *next;
6  *     ListNode(int x) : val(x), next(NULL) {}
7  * };
8  */
9 class Solution {
10 public:
11     ListNode* deleteDuplicates(ListNode* head) {
12         ListNode* dummyHead = new ListNode(0);
13         dummyHead->next = head;
14         ListNode* validatedPtr = dummyHead;
15
16         while(head!=nullptr){
17             while(head->next!=nullptr && head->next->val == head->val)
18                 head = head->next;
19             validatedPtr->next = head;
20             validatedPtr = validatedPtr->next;
21             head = head->next;
22         }
23         validatedPtr->next = nullptr;
24         return dummyHead->next;
25     }
26 };
```