83. Remove Duplicates from Sorted List

⤭ Pick One

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

Example 1:

```
Input: 1->1->2
Output: 1->2
```

Example 2:

```
Input: 1->1->2->3->3
Output: 1->2->3
```

```java
public class L83 {
    public class ListNode {
            int val;
            ListNode next;
            ListNode(int x) { val = x; }
        }
    /*
     *  这道题目是去除链表中的重复元素
     */
    public ListNode deleteDuplicates(ListNode head) {
            if (head == null || head.next == null) {
                return head;
            }
            //记住用一个preNode，不要利用原来的head
            ListNode preNode = new ListNode(0);
            preNode.next = head;

            ListNode node = head;
            ListNode next = head.next;
            //指针变换
            while (next != null) {
                if(node.val != next.val) {
                    node.next = next;
                    node = next;
                    next = next.next;
                }else
                    next = next.next;
            }
        //记住这儿要置空。[1,2,2,3,3]否则遇到这种情况，最后的结果是[1,2,3,3]，这时候node为第一个3，next为第二个3。
        //置空的时候就是去掉next
            node.next = null;
            return preNode.next;
    }
}
```