//Assignment 11

//23.

class Solution {

public ListNode mergeKLists(ListNode[] lists) {

     List<Integer> values = new ArrayList<>();

       for (int i = 0; i < lists.length; i++) {

    ListNode list = lists[i];

            while (list != null) {

                values.add(list.val);

                list = list.next;

            }

        }

        Collections.sort(values);

        ListNode dummy = new ListNode(0);

        ListNode current = dummy;

        for (int i = 0; i < values.size(); i++) {

          int val = values.get(i);

            current.next = new ListNode(val);

            current = current.next;

        }

        return dummy.next;

    }

}

//234.

class Solution {

    public void deleteNode(ListNode node) {

        node.val = node.next.val;

        node.next = node.next.next;

    }

}

//83.

class Solution {

    public ListNode deleteDuplicates(ListNode head) {

        ListNode current=head;

        while(current!=null&&current.next!=null){

            if(current.val==current.next.val){

                current.next=current.next.next;

            }

            else{

                current=current.next;

            }

        }

        return head;

    }

}