23)

class Solution {

    public ListNode mergeKLists(ListNode[] lists) {

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

        for (ListNode head : lists) {

            while (head != null) {

                values.add(head.val);

                head = head.next;

            }

        }

        Collections.sort(values);

        ListNode dummy = new ListNode(0);

        ListNode c = dummy;

        for (int val : values) {

            c.next = new ListNode(val);

            c = c.next;

      }

        return dummy.next;

    }

}

237)

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 temp=head;

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

            while(temp.next!=null&& temp.val==temp.next.val){

                temp.next=temp.next.next;

            }

            temp=temp.next;

        }

        return head;

    }

}