You are given an array of k linked-lists lists, each linked-list is sorted in ascending order.*Merge all the linked-lists into one sorted linked-list and return it.*

**Example 1:**

**Input:** lists = [[1,4,5],[1,3,4],[2,6]]

**Output:** [1,1,2,3,4,4,5,6]

**Explanation:** The linked-lists are:

[

1->4->5,

1->3->4,

2->6

]

merging them into one sorted list:

1->1->2->3->4->4->5->6

**Constraints:**k == lists.length

* 0 <= k <= 104
* 0 <= lists[i].length <= 500
* -104 <= lists[i][j] <= 104
* lists[i] is sorted in **ascending order**.
* The sum of lists[i].length will not exceed 104.

Solution :

class Solution {

public:

    ListNode\* mergeKLists(vector<ListNode\*>& lists) {

        priority\_queue<int,vector<int>,greater<int>>pq;

        if(lists.size()==0) return NULL;

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

            ListNode\* p=lists[i];

            while(p){

                pq.push(p->val);

                p=p->next;}        }

        if(pq.size()==0) return NULL;

        ListNode\* head = new ListNode;

        head->val=pq.top();

        head->next=NULL;

        pq.pop();

        ListNode\* p=head;

        while(!pq.empty()){

            ListNode\* tmp=new ListNode;

            tmp->val=pq.top();

            pq.pop();

            tmp->next=NULL;

            p->next=tmp;

            p=p->next;       }

        return head;

    }

};