61Rotate List

Given a list, rotate the list to the right by *k* places, where *k* is non-negative.

Example:

```
Given 1->2->3->4->5->NULL and k = 2,

return 4->5->1->2->3->NULL.
```

【思路】

可以遍历链表得到链表长度，首尾相接形成环形链表。k可以大于链表的长度，所以需要取模。

```
/**
 * Definition for singly-linked list.
 * struct ListNode {
 *     int val;
 *     ListNode *next;
 *     ListNode(int x) : val(x), next(NULL) {}
 * };
 */
class Solution {
public:
    ListNode* rotateRight(ListNode* head, int k) {
        if(head==NULL)  return head;
        ListNode* node = head;
        int cnt=1;
        while(node->next){
            node = node->next;
            cnt++;
        }
        if(cnt!=0) k = k%cnt;
        node->next = head;
        for(int i =0;i<cnt-k;i++){
            node = node->next;
        }
        ListNode* n_head = node->next;
        node->next = NULL;
        return n_head;
    }
};
```