Merge Sorted Lists

Question: Merge two sorted linked lists and return it as a new list. The new list should be made by splicing together the nodes of the first two lists.

Solutions:

```
class ListNode:
    def __init__(self, x):
        self.val = x
        self.next = None


class Solution:
    # @param two ListNodes
    # @return a ListNode
    def mergeTwoLists(self, l1, l2):
        dummy = ListNode(0)
        pointer = dummy
        while l1 !=None and l2 !=None:
            if l1.val<l2.val:
                pointer.next = l1
                l1 = l1.next
            else:
                pointer.next = l2
                l2 = l2.next
            pointer = pointer.next
        if l1 == None:
            pointer.next = l2
        else:
            pointer.next = l1
        return dummy.next

    def printll(self, node):
        while node:
            print ( node.val )
            node = node.next

if __name__ == '__main__':
    ll1, ll1.next, ll1.next.next = ListNode(2), ListNode(3), ListNode(5)
    ll2, ll2.next, ll2.next.next = ListNode(4), ListNode(7), ListNode(15)
    Solution().printll( Solution().mergeTwoLists(ll1,ll2) )
```