```java
/**
 * Definition for singly-linked list.
 * public class ListNode {
 *     int val;
 *     ListNode next;
 *     ListNode() {}
 *     ListNode(int val) { this.val = val; }
 *     ListNode(int val, ListNode next) { this.val = val; this.next = next; }
 * }
 */
class Solution {
    public ListNode addTwoNumbers(ListNode l1, ListNode l2) {
        ListNode dummy = new ListNode(0); // creating an dummy list
        ListNode curr = dummy; // intialising an pointer
        int carry = 0; // intialising our carry with 0 intiall
        // while loop will run, until l1 OR l2 not reaches null OR if they both reaches null. But our carry has some value in it.
        // We will add that as well into our list
        while(l1 != null || l2 != null || carry == 1){
            int sum = 0; // intialising our sum
            if(l1 != null){ // adding l1 to our sum & moving l1
                sum += l1.val;
                l1 = l1.next;
            }
            if(l2 != null){ // adding l2 to our sum & moving l2
                sum += l2.val;
                l2 = l2.next;
            }
            sum += carry; // if we have carry then add it into our sum
            carry = sum/10; // if we get carry, then divide it by 10 to get the carry
            ListNode node = new ListNode(sum % 10); // the value we'll get by moduloing it, will become as new node so. add it to our list
            curr.next = node; // curr will point to that new node if we get
            curr = curr.next; // update the current every time
        }
        return dummy.next; // return dummy.next bcz, we don't want the value we have consider in it intially!!
    }
}
```