Problem-1 : Add Two Numbers

```
package com. ishwarchavan;

class Solution {
    public ListNode addTwoNumbers(ListNode l1, ListNode l2) {
                //    passing two listnode


    ListNode dummy = new ListNode(0);
      //initializing  and declaring all the needed variable here
    ListNode p=11, q=12, curr=dummy;
    int carry=0;                                                                    //
assing 0 valur to the carry

    while(p !=null || q !=null) {                                   //checking while
condition , if satisfied then it will executed
      int x=p !=null ? p.val : 0;                                           //It will check
the size of Listnodes with condition operator
      int y=q !=null ? q.val : 0;
      int sum = x+y +carry;                                                 //adding two
varialble x & y
      carry =sum/10;                                    //Getting carry value by dividing
sum by 10

      curr.next = new ListNode(sum%0);    //creating new node which s attache with
curr.node

      curr = curr.next;                             //shifting the node
      if(p !=null) p = p.next;                      //shiftig all the pointer with node
      if(q !=null) q = q.next;
    }
    if(carry > 0) {                                                 //satisfied the
condition then we need to attach the current carry value
      curr.next = new Listnode(carry);
    }
      return dummy.next;                            // return the value

    }
}
```

2. Palindrome

```
package com.ishwarchavan;

class Solution{
      public boolean isPalindrome(int x) {            //creating function is Palindrome
with integr value
            if (x<0) {                                          //condition if x is
negative value then it will false
                  return false;

      int reverseOfx =0;                                //taking int variable to store
result vlaue
                  int copyOfx=x;                                //initialize the value to
the copyofx variable

                  while(copyOfx !=0){                           //condition is checking
                        reverseOfx = (reverseOfx * 10) + (copyOfx %10);
      //adding both value with remainder

                        copyOfx = copyOfx/10;
      }
      return reverseOfx == x;       //checking the original value is same or not
            }
      }
}
```