**Solution 1 (even - odd)**

def checkEvenOrOdd(number):

    if number % 2 == 0:

        print("True")

    else:

        print("False")

# number = 12

number = 11

checkEvenOrOdd(number)

**Solution 2 (palindrome)**

def checkPalindrome(number):

    copy\_num = number

    sum = 0

    while copy\_num > 0:

        reminder = copy\_num%10

        sum = sum \* 10 + reminder

        copy\_num = copy\_num // 10

    return sum

number = 121

ans = checkPalindrome(number)

if ans == number:

    print('true')

else:

    print('false')

**Solution 3**

def findUnique(arr, n):

     # O(n\*n)  brute force approach

    for i in range(n):

        count = 0

        for j in range(n):

            if arr[i] == arr[j]:

                count += 1

        if count == 1:

            return arr[i]

    return -1

**Best Approaches of unique value**

    # O(n) time complexity using dictionary

    # hashing

    data = {}

    for i in arr:

        data[i] = data.get(i,0) + 1

    for i in data:

        if data[i] == 1:

            return i

    # O(n) time complexity

    ans = 0

    for i in arr:

        ans = ans ^ i

    return ans

arr = [1,2,3,1,2]

n= len(arr)

ans = findUnique(arr,n)

print(ans)

**Solution 4 (0-1 sort)**

def sortZeroesAndOne(arr, n) :

    one = 0

    for i in range(n):

        if arr[i] == 0:

            arr[i],arr[one] = arr[one],arr[i]

            one += 1

arr = [0,1,0,0,1,1,0]

n = len(arr)

sortZeroesAndOne(arr, n)

print(arr)

**Solution 5 majority element**

**Brute Force solution O(N\*N)**

def majorityElement(nums):

    count = 0

    n = len(nums)

    for num in nums:

        count = nums.count(num)

        if count >= (n//2):

            return num

    return -1

arr = [1,2,1,1,1,2]

ans = majorityElement(arr)

print(ans)

**Best Solution in O(n)**

def majorityElement(nums):

    ans = nums[0]

    count = 0

    for num in nums:

        if num == ans:

            count += 1

        elif count == 0:

            ans = num

            count = 1

        else:

            count -= 1

    return ans

**Solution 6**

def reverseString(word):

    ans = ""

    n = len(word)

    for latter in range(n-1,-1,-1):

        ans += word[latter]

    return ans

word = "hello"

ans = reverseString(word)

print(ans)

7.

def removeConsecutiveDuplicates(string) :

    if len(string) == 0:

        return ""

    ans = ""

    for i in range(len(string)-1):

        if string[i] != string[i+1]:

            ans += string[i]

    ans += string[-1]

    return ans

string = “aaaa”

ans = removeConsecutiveDuplicates(string)

print(ans)