```python
# 1. Write a Python Program to Count and Filter Odd and Even Numbers
of Given List Using Loops.

l2=[1,4,5,7,2,9]
l_odd=[]
l_even=[]
count_even=0
for i in l2:
   if i%2==0:
      count_even=count_even+1
      l_even.append(i)
   else:
       l_odd.append(i)

print(f"List of Even numbers {l_even} and count {count_even}")
print(f"List of odd numbers {l_odd} and count {len(l2)-count_even}")
```

```
List of Even numbers [4, 2] and count 2
List of odd numbers [1, 5, 7, 9] and count 4
```

```python
#Write a Python Program to find all possible combinations of a List
with three elements without using builtin function.
l2=[1,2,3]
for i in l2:
   for j in l2:
       for k in l2:
           if i!=j and j!=k and i!=k:
               print(i,k,j)
```

```
1 3 2
1 2 3
2 3 1
2 1 3
3 2 1
3 1 2
```

```python
#Python program to print all Strong(maximum for every adjacent pair)
numbers in given list.
l2 =[1,4,5,6,0,2,9]
for i in range(len(l2)-1):
    print(max(l2[i],l2[i+1]))
```

```
4
5
6
6
2
9
```

```
#Maximum sum of elements of a list in a list of lists. Sample Input: [[3, 4, 5], [1, 2, 3], [0, 9, 0]]. Output: 12
l2= [[3, 4, 5], [1, 2, 3], [0, 9, 0]]
sum2=-2**31;
for i in l2:
  sum2=max(sum2,sum(i))

print(sum2)
```

```
12
```

```
#Python program to find number of m contiguous elements of a List with a given sum(Given a list 'L', a sum 'S' and number of elements to take at a time 'm". The task is to find how many ways sum s can be found by adding any m contiguous elements).

l2=[1,2,3,4,5,6]
sum3=9
m=2
cal_sum=0
count=0
for i in range(len(l2)-2):
    k=i
    for k in range(m):
        cal_sum+=l2[k]
        if cal_sum == sum3:
            count+=1;

print(count)
```

```
1
```

```
#Python Program to accept the strings which contains all vowels (Given a string the task is to check if every vowel is present or not. We consider a vowel to be present if it is present in upper case or lower case. i.e. 'a', 'e', 'i', 'o', 'u' or 'A', E', 'I', 'O', 'U'
str="i am elephant uo"
l2=[0,0,0,0,0]
for i in range(len(str)):
    if str[i]=='a' or str[i]=='A' and l2[0]==0:
        l2[0]=1;
    elif str[i]=='e' or str[i]=='E' and l2[2]==0:
        l2[1]=1;
    elif str[i]=='i' or str[i]=='I' and l2[2]==0:
        l2[2]=1;
    elif str[i]=='o' or str[i]=='O' and l2[3]==0:
        l2[3]=1;
    elif str[i]=='u' or str[i]=='U' and l2[4]==0:
        l2[4]=1;
```

```
if sum(l2) == 5:
    print("yes")
else:
    print("NO")
```

```
yes
```

```
#Write a program to sort list of tuple based on sum(without using a
built in function)

li=[(3,4),(6,7),(12,-2),(90,-91)]
total=0
l5=[]
l6=[]
for i in li:
   l5.append(sum(i))

l5.sort()

for i in l5:
    for j in li:
        if i==sum(j):
            l6.append(j)

print(l6)
```

```
[(90, -91), (3, 4), (12, -2), (6, 7)]
```

```
#Python program to construct a n*m matrix from list. Write a program
to do the Matrix addition using nested lists
matrix = []
matrix1=[]
for i in range(2):
    l2 = []
    for j in range(3):
        val = int(input("Enter element: "))
        l2.append(val)
    matrix.append(l2)

for i in range(2):
    l2 = []
    for j in range(3):
        val = int(input("Enter element: "))
        l2.append(val)
    matrix1.append(l2)


n = len(matrix)
m = len(matrix[0])
result = []
for i in range(n):
```

```
    row = []
    for j in range(m):
         row.append(matrix[i][j] + matrix1[i][j])
    result.append(row)

print("Matrix Sum :")
print(result)
```

```
Enter element:  1
Enter element:  2
Enter element:  3
Enter element:  4
Enter element:  5
Enter element:  6
Enter element:  7
Enter element:  8
Enter element:  9
Enter element:  10
Enter element:  11
Enter element:  12

Matrix Sum :
[[8, 10, 12], [14, 16, 18]]
```