```
//DSL Lab 02 Operation on String


#input conversion
def string_list(string):
    strlist = string.split(' ')
    return strlist

#count element in lis
def count(lis):
    N = 0
    for i in lis:
        N = N +1
    #print(N)
    return N

#longest string
def longest(lis):
    string = lis[0]
    #length = len(lis[0])
    for i in range(len(lis)):
        if len(lis[i])>len(string):
            string = lis[i]

    print("Longest word in string is: ",string)

#frequency of character
def frequency_char(string):
    c = input("Enter character: ")
    f = 0
    for i in string:
        if i == c:
            f = f + 1
    print("Frequency of",c,"in",string,"is",f)
#remove duplicate
def removeDupli(lis):
     lst=[]
     for i in lis:
        if i not in lst:
            lst.append(i)
     return lst

#count of substring
def count_str(lis):
    ans = []
    f = 0
    while(f<len(lis)):
        c = 0
        ele = lis[f]
        for i in lis:
            if i == ele:
                c = c + 1
        f = f+1
        ans.append(str(ele)+":"+str(c))
        #print(ele,c)
```

```
        c=0
    print("String : Frequency is ",removeDupli(ans))

#palindrome or not
def is_palindrome(string):
     if string == string[::-1]:
         print("String ",string,"is palindrome.")
     else:
        print("String",string,"is not palindrome.")

#index of first letter of substring

def index_substring(string):
   substring = input("Enter a substring: ")
  # i = string.find(substring)
   for i in range(0,len(string)):
       if(string[i:len(substring)] in substring):
           print("Index of first letter of substring is: ",i)
           break

#main function
string = input("Enter any string: ")

otp = 24
while(otp !=0):
     print("MAIN MENU \n1.To display word with the longest length\n2.To
determines the frequency of occurrence of particular character in the
string\n3.To check whether given string is palindrome or not\n4.To display
index of first appearance of the substring\n5.To count the occurrences of
each word in a given string")
     otp = int(input("Enter input for operation(0 to exit): "))

     if(otp==1):
        longest(string_list(string))
     elif(otp==2):
        frequency_char(string)
     elif(otp==3):
        is_palindrome(string)
     elif(otp==4):
        index_substring(string)
     elif(otp==5):
        count_str(string_list(string))tage_lis[-i])
```