In [ ]:
```
#Running instructions
'''shift+enter --->Run cell
   Ctrl+s      --->Save
   Ctrl+M+B    --->New cell
   Ctrl+M+D    --->Delete cell
   Shift+/     --->Comment'''
```

In [1]:
```
#Correcting an error
 n=int(input("Enter Number"))
if n%2==0:
    print("Its is even number")
else:
print('Its odd number')
```

```
  File "<ipython-input-1-3088f3913a7a>", line 2
    n=int(input("Enter Number"))
    ^
IndentationError: unexpected indent
```

In [2]:
```
n=int(input("Enter Number"))
if n%2==0:
    print("Its is even number")
else:
    print('Its odd number')
```

```
Enter Number3
Its odd number
```

In [4]:
```
#Exercise 2: 3.GCD OF TWO NUMBERS
def gcd(a,b):
    if a==0:
        return b
    return gcd(b%a,a)
a=int(input('Enter first number:'))
b=int(input('Enter second number:'))
res1=gcd(a,b)
print(res1)
```

```
Enter first number:7
Enter second number:8
1
```

In [9]:
```
# 4.add.py TAKES NUMBERS FROM COMMMAND PROMPT AND EXECUTE
import sys
x=int(sys.argv[1])
y=int(sys.argv[2])
sum=x+y
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-9-176e7ad72241> in <module>
      1 # 4.add.py TAKES NUMBERS FROM COMMMAND PROMPT AND EXECUTE
      2 import sys
----> 3 x=int(sys.argv[1])
      4 y=int(sys.argv[2])
      5 sum=x+y

ValueError: invalid literal for int() with base 10: '-f'
```

In [11]:
```
# EXERCISE 3- 5.GIVE NUMBER IS EVEN OR NOT
n=int(input())
if(n%2==0):
    print("It is a even number")
else:
    print("It is a odd number")
```

```
44
It is a even number
```

In [12]:
```
#6.USING FOR LOOP SHOW OVER A SEQUENCE
n=int(input("Enter the number"))
for i in range(n):
    print(i,end=" ")
```

```
Enter the number7
0 1 2 3 4 5 6
```

In [15]:
```
# 7.PRINT FIBONACCI SERICES USING WHILE LOOP
n=int(input("Enter number of terms:"))
count=0
n1=0
n2=1
while count<n:
    print(n1)
    temp=n1+n2
    n1=n2
    n2=temp
    count+=1
```

```
Enter number of terms:5
0
1
1
2
3
```

In [17]:
```
# 8.print all prime numbers using break
n,m=map(int,input("Enter the Range ").split())
for i in range(n,m):
    if i>1:
        for j in range(2,i):
            if(i%j)==0:
                break
        else:
                print(i,end=" ")
```

```
Enter the Range 0 50
2 3 5 7 11 13 17 19 23 29 31 37 41 43 47
```

In [19]:
```
# EXERCISE 4 : 9 . Mean,Median,Mode of a given set
l=[2,33,4,5,7,44,2]
mean=sum(l)/len(l)
median=l[len(l)//2]
mode=max(set(l),key=l.count)
print(mean)
print(median)
print(mode)
```

```
13.857142857142858
5
2
```

In [20]:
```
# 10. Convert list and tuple into array
import numpy as np
li=[2,3,5,7,99,4,5]
tup=(4,88,1,6,90,3,10)
arr=np.array(li)
arr1=np.array(tup)
print(arr)
print(arr1)
```

```
[ 2  3  5  7 99  4  5]
[ 4 88  1  6 90  3 10]
```

In [23]:
```
# 11. Find common values between two arrays
a=[int(x) for x in input().split()]
b=[int(x) for x in input().split()]
common=[]
for i in a:
    if i in b:
        common.append(i)
print(set(common))
```

```
1 2 3
4 3 2
{2, 3}
```

In [24]:
```
# EXERXCISE 5. 12 : Count number of characters in a string and store them in a dictionary data structure
dic={}
s=input()
sl=list(s)
for i in sl:
    if i not in dic:
        dic[i]=1
    else:
        dic[i]=dic[i]+1
print(dic)
```

```
karthik
{'k': 2, 'a': 1, 'r': 1, 't': 1, 'h': 1, 'i': 1}
```

In [25]:
```
# 13.Combine list into dictionary
a=[(x) for x in input("Enter first list:").split()]
b=[int(x) for x in input("Enter the second list:").split()]
dic={}
for i in a:
    for j in b:
        dic[i]=j
        b.remove(j)
        break
print(dic)
```

```
Enter first list:a b c d
Enter the second list:1 2 3 4
{'a': 1, 'b': 2, 'c': 3, 'd': 4}
```

In [27]:
```
# Exercise 6: 14.String starts with specified characters
s=input("Enter the string: ")
r=input("Enter the character:")
if(s[0]==r):
    print("YES! It is starts with a specificed character")
else:
    print("No!! It didn't started")
```

```
Enter the string: karthik
Enter the character:k
YES! It is starts with a specificed character
```

In [29]:
```
# 15.Check whether a string is palindrome or not
s=input("Enter the String: ")
s=s.lower()
rev=s[::-1]
if(s==rev):
    print("It is palindrome")
else:
    print("it is not palindrome")
```

```
Enter the Stringmom
It is palindrome
```

In [32]:
```
# Exercise 7 : 16 . Split and join a string
s=input("Enter the string: ")
l=s.split(" ")
res=".".join(l)
print(res)
```

```
Enter the string: hi this is karthik
hi.this.is.karthik
```

In [33]:
```
# 17. Sort the words in Alphabetic orders
a=input("Enter the string: ")
l=a.split(" ")
s=sorted(l)
print(s)
```

```
Enter the string: hi where is vishnu
['hi', 'is', 'vishnu', 'where']
```

In [2]:
```
#Exercise 8 : 18. Print each line of a file in reverse order
f=open("file-46.txt","r")
s=""
for i in f:
    s=s+i[::-1]
print("Reverse of a string : ",s)
f.close()
```

```
Reverse of a string :
.kihtraK si eman yM
.tnemtraped ecneicS ataD morf ma I .5202 no egelloc TEIJV RNV morf tuo ssap lliw I
```

In [4]:
```
# 19 . Compute Number of lines,words and characters in a file
f=open("file-46.txt","r")
s=""
for i in f:
    s=s+i
    print("Number of characters: ",len(s)-s.count(''))
    print("Number of words: ",s.count(''))
    print("Number of lines: ",s.count('\n'))
```

```
Number of characters:  -1
Number of words:  21
Number of lines:  1
Number of characters:  -1
Number of words:  56
Number of lines:  2
Number of characters:  -1
Number of words:  104
Number of lines:  2
```

In [8]:
```
# 20 . Count the frequency of characters in a file
f=open("file-46.txt","r")
s=""
for i in f:
    s=s+i
    d={}
for i in s:
        if i in d:
            d[i]+=1
        else:
            d[i]=1
print(d)
```

```
{'M': 1, 'y': 1, ' ': 18, 'n': 4, 'a': 7, 'm': 5, 'e': 7, 'i': 4, 's': 3, 'K': 1, 'r': 4, 't': 5, 'h': 1, 'k': 1, '.': 3, '\n': 2, 'I': 3, 'f': 2, 'o': 5,
'D': 1, 'S': 1, 'c': 3, 'd': 1, 'p': 2, 'w': 1, 'l': 4, 'u': 1, 'V': 2, 'N': 1, 'R': 1, 'J': 1, 'E': 1, 'T': 1, 'g': 1, '2': 2, '0': 1, '5': 1}
```

In [38]:
```
#Exercise 9: 21 Calculator using functions
def add(x,y):
    print(x+y)
def sub(x,y):
    print(x-y)
def mul(x,y):
    print(x*y)
def div(x,y):
    print(x/y)
print("1.Addition\n 2.Subtraction\n 3.Multiplication\n 4.Division\n")
n=int(input())
x=int(input("Enter the first value: "))
y=int(input("Enter the second value: "))
if n==1:
    add(x,y)
elif n==2:
    sub(x,y)
elif n==3:
    mul(x,y)
elif n==4:
    div(x,y)
else:
    print("No Option")
```

```
1.Addition
 2.Subtraction
 3.Multiplication
 4.Division

1
Enter the first value: 3
Enter the second value: 4
7
```

In [40]:
```
# 22.Factorial using recursion
def fact(n):
    if(n==0):
        return 1
    elif(n==1):
        return 1
    else:
        return n*fact(n-1)
n=int(input("Enter the value: "))
result=fact(n)
print(result)
```

```
Enter the value: 5
120
```

In [44]:
```
#23.Find duplicates in the list using functions
def dup(l):
    a=set(l)
    for i in a:
        if(l.count(i)>1):
            res.append(i)
        return res
    b=[int(x) for x in input().split()]
    res=[]
    print(dup(b))
```

In [49]:
```
# 24. Find unique elements in the list
def uni(lis):
    li=set(lis)
    for i in li:
        if(lis.count(i)==1):
             res_li.append(i)
    return res_li
li=[int(x) for x in input().split()]
res_li=[]
result=uni(li)
print(result)
```

```
1 2 3 4 2 3
[1, 4]
```

In [53]:
```
# 25.Cumulative product of a list of numbers
def cum(lis):
    mul=1
    for i in lis:
        mul=i*mul
    return mul
b=[int(x) for x in input().split()]
result=cum(b)
print(result)
```

```
10
10
```

In [54]:
```
# 26.print reverese order of a list
def rev(lis):
    org=lis
    for i in range(len(org)):
        rev_lis.append(lis.pop())
    return rev_lis
b=[int(x) for x in input().split()]
rev_lis=[]
result=rev(b)
print(result)
```

```
4 6 7  8 2 1
[1, 2, 8, 7, 6, 4]
```

In [56]:
```
# 27.Compute gcd,lcm of 2 numbers
def gcd(a,b):
    if a==0:
        return b
    return gcd(b%a,a)
def lcm(res1):
    return (a//res1)*b
a=int(input("Enter first number: "))
b=int(input("Enter second number: "))
res1=gcd(a,b)
res2=lcm(res1)
print(res2,res1)
```

```
Enter first number: 4
Enter second number: 3
12 1
```

In [ ]: