In [3]:

```
a=3
type(a)
```

Out[3]:

```
int
```

In [4]:

```
a=3.576
type(a)
```

Out[4]:

```
float
```

In [5]:

```
a="Apple"
type(a)
```

Out[5]:

```
str
```

In [6]:

```
a={"b":"a","na":"na"}
type(a)
```

Out[6]:

```
dict
```

In [7]:

```
a=(1,"orange",32,"cat",67.8902)
type(a)
```

Out[7]:

```
tuple
```

In [11]:

```
a=[1,2,3,4]
type(a)
```

Out[11]:

```
list
```

In [9]:

```
a={1,2,3}
type(a)
```

Out[9]:

```
set
```

In [10]:

```
a=4+6
b=8-2
c=7*9
d=6/2
e=3**7
f=22//8
g=4%2
print(a,b,c,d,e,f,g)
```

```
10 6 63 3.0 2187 2 0
```

In [12]:

```
a=3>2
b=3<7
c= 6==7
d=5!=5
e=9>=10
f=9<=15
print(a,b,c,d,e,f)
```

```
True True False False False True
```

In [13]:

```
a=float(input("enter the base:"))
b=float(input("enter the height:"))
c=0.5*a*b
print("the area of triangle",c)
```

```
enter the base:8
enter the height:5
the area of triangle 20.0
```

In [14]:

```
a=6
b=9
if(a==b):
print(same)
```

```
  File "C:\Users\dell\AppData\Local\Temp/ipykernel_18648/2311583190.py", line 4
    print(same)
    ^
IndentationError: expected an indented block
```

In [12]:

```
a=int(input("enter a number: "))
b=int(input("enter a number: "))
if(a>b):
    greater=a
else:
    greater=b
for x in range(1,greater+1):
    if((a%x==0) and (b%x==0)):
        gcd=x
print("GCD of",a,"and",b,"is",gcd)
```

```
enter a number: 5
enter a number: 25
GCD of 5 and 25 is 5
```

In [13]:

```
a=int(input("enter a number: "))
b=int(input("enter a number: "))
if(a>b):
    smaller=b
else:
    smaller=a
for x in range(1,smaller+1):
    if((a%x==0) and (b%x==0)):
        gcd=x
print("GCD of",a,"and",b,"is",gcd)
```

```
enter a number: 2
enter a number: 48
GCD of 2 and 48 is 2
```

# Even or not

In [14]:

```
num=int(input("enter a number: "))
if(num%2==0):
    print(num,"is an even number")
else:
    print(num,"is not an even number")
```

```
enter a number: 5
5 is not an even number
```

# for loop

In [16]:

```
for x in range(0,11):
    print(x)
```

```
0
1
2
3
4
5
6
7
8
9
10
```

# Fibonacci sequence

In [22]:

```
n=int(input("enter the number of terms: "))
a=0
b=1
print(0,end=" ")
for i in range(1,n+1):
    print(b,end=" ")
    x=a+b
    a=b
    b=x
```

```
enter the number of terms: 10
0 1 1 2 3 5 8 13 21 34 55
```

# Fibonacci sequence using while loop

In [18]:

```
n=int(input("enter the number of terms: "))
a=0
b=1
i=0
print(0, end=" ")
if(n<0):
    print("Enter a positive number")
while(i<n):
    print(b, end=" ")
    x=a+b
    a=b
    b=x
    i=i+1
```

```
enter the number of terms: 10
0 1 1 2 3 5 8 13 21 34 55
```

# Mean,Median,Mode

In [6]:

```
list3=[]
n=int(input("enter the number of elements in a list: "))
for i in range(0,n):
    list3.append(int(input()))
list3.sort()
print(list3)
total=0
for x in list3:
    total=total+x
mean=total/n
print("The mean of the given data is",mean)
if(n%2!=0):
    median=n//2
    print("Median of the given data is",list3[median])
else:
    median=list3[n//2]+list3[(n-1)//2]
print("Median of the given data is",median/2)
mode=list3[0]
count=0
for i in list3:
    a=list3.count(i)
    if a>count:
        count=a
        mode=i
print("Mode of the given data is:",mode)
```

```
enter the number of elements in a list: 6
2
4
6
8
1
12
[1, 2, 4, 6, 8, 12]
The mean of the given data is 5.5
Median of the given data is 5.0
Mode of the given data is: 1
```

# List and Tuple into Array

In [1]:

```
import numpy as np
list=[10,12,13,14,15,16,17,18]
arr1=np.array(list)
tuple=(2,4,6,8,10,122)
arr2=np.array(tuple)
print(arr1,arr2)
```

```
[10 12 13 14 15 16 17 18] [  2   4   6   8  10 122]
```

# Common values between two arrays

In [2]:

```
import numpy as np
array1=[0,10,20,40,60]
array2=[10,20,50,82]
print(np.intersect1d(array1,array2))
```

```
[10 20]
```

In [1]:

```
s=input("Enter the name: ")
f={}
for i in s:
    if i in f:
        f[i]+=1
    else:
        f[i]=1
print("Number of characters in string is: ",f)
```

```
Enter the name: banana
Number of characters in string is:  {'b': 1, 'a': 3, 'n': 2}
```

In [2]:

```
a=["name","age","height"]
b=["Shiva",25,"6 feet"]
c={}
for i in a:
    for j in b:
        c[i]=j
        b.remove(j)
        break
print(c)
```

```
{'name': 'Shiva', 'age': 25, 'height': '6 feet'}
```

String starting with special character

In [4]:

```
a=input("Enter your string: ")
spl=input("Enter your special character: ")
if a==spl:
    print("It starts with special character")
else:
    print("It does not start with special character")
```

```
Enter your string: albert
Enter your special character: A
It does not start with special character
```

Palindrome or not

In [4]:

```
a=input("Enter a string: ")
b=a[::-1]
if(a==b):
    print("It is palindrome")
else:
     print("It is palindrome")
```

```
Enter a string: nana
It is not a palindrome
```

Symmetric or not

In [4]:

```
a=input("Enter a string: ")
n=len(a)
if(n%2==0):
    for i in range(0,int((n/2))):
        x=a[i]
        for i in range(int((n/2)),int(n)):
            y=a[i]
        if(x==y):
            print(a,"is symmetric")
        else:
            print(a,"is not symmetric")
        break
else:
    print(a,"is not symmetric")
```

```
Enter a string: checkclark
checkclark is not symmetric
```

Anagram or not

In [7]:

```
a="pramod"
b="rapmod"
if(sorted(a.lower())==sorted(b.lower())):
    print(a,b,"are anagram")
else:
    print(a,b,"are not anagram")
```

```
pramod rapmod are anagram
```

# Split & join

In [1]:

```
fact="Ramu is a good boy"
n=fact.split()
print(n)
```

```
['Ramu', 'is', 'a', 'good', 'boy']
```

In [2]:

```
b=["It's","true","Ramu","is","a","good","boy"]
n=" ".join(b)
print(n)
```

```
It's true Ramu is a good boy
```

# Sorting

In [3]:

```
f="ball cat apple"
x=f.split()
x.sort()
for y in x:
    print(y)
```

```
apple
ball
cat
```

In [4]:

```
fact="apple"
print(sorted(fact))
```

```
['a', 'e', 'l', 'p', 'p']
```

# file basics

In [58]:

```
file=open("C:/Users/admin/Desktop/python.txt","a")
file.write("Hope you are good in python")
file.close()
print(file)
```

```
<_io.TextIOWrapper name='C:/Users/admin/Desktop/python.txt' mode='a' encodin
g='cp1252'>
```

In [59]:

```
file=open("C:/Users/admin/Desktop/python.txt","a")
file.write(" what have you learnt in it")
file.close()
print(file)
```

```
<_io.TextIOWrapper name='C:/Users/admin/Desktop/python.txt' mode='a' encodin
g='cp1252'>
```

In [60]:

```
file=open("C:/Users/admin/Desktop/python.txt","r")
print(file.read())
```

```
Hope you are good in python what have you learnt in it
```

# print each line of a file in reverse order

In [82]:

```
file=open("C:/Users/admin/Desktop/python.txt","r")
s=""
for i in file:
    s=s+i
    print(s[::-1])
file.close()
```

```
ti ni tnrael uoy evah tahw nohtyp ni doog era uoy epoH
```

# compute the number of characters, words and lines in a file

In [83]:

```
file=open("C:/Users/admin/Desktop/python.txt","r")
s=""
for i in file:
    s=s+i
print("Number of characters = ",len(s)-s.count(' ')-s.count('\n'))
print("Number of words = ",s.count(' ')+s.count('\n')+1)
print("Number of lines = ",s.count('\n')+1)
file.close()
```

```
Number of characters =  43
Number of words =  12
Number of lines =  1
```

# count frequency of characters in a given file.

In [99]:

```
file=open("C:/Users/admin/Desktop/python.txt","r")
s=""
fre={}
for i in file:
    s=s+i
for j in s:
    if j in fre:
        fre[j]+=1
    else:
        fre[j]=1
print(s)
print("Fequency of characters in the file is:\n",fre)
file.close()
```

```
Hope you are good in python what have you learnt in it
Fequency of characters in the file is:
 {'H': 1, 'o': 6, 'p': 2, 'e': 4, ' ': 11, 'y': 3, 'u': 2, 'a': 4, 'r': 2,
'g': 1, 'd': 1, 'i': 3, 'n': 4, 't': 4, 'h': 3, 'w': 1, 'v': 1, 'l': 1}
```

```
10.1-Write a function cumulative_ product to compute cumulative product of a list of
numbers
```

In [16]:

```
def cp(a):
    b=[]
    k=1
    for i in range(0,len(a)):
        k=a[i]*k
        b.append(k)
    print(b)
    print(k)
a=[1,2,3,4,1,6,2]
cp(a)
```

```
[1, 2, 6, 24, 24, 144, 288]
288
```

```
10.2-Write a function reverse to print the given list in the reverse order.
```

In [1]:

```
def rev(a):
    print(a[::-1])

a=[1,2,3,4,1,6,2]
rev(a)
```

```
[2, 6, 1, 4, 3, 2, 1]
```

10.3-Write function to compute GCD, LCM of two numbers

In [1]:

```
import math
def gcd(a,b):
    if(a>b):
        smaller=b
    else:
        smaller=a
    for x in range(1,smaller+1):
        if((a%x==0) and (b%x==0)):
            gcd=x
    print("GCD of",a,"and",b,"is",gcd)
def lcm(a,b):
    print("LCM of",a,"and",b,"is",math.lcm(a,b))

a=int(input("Enter a number: "))
b=int(input("Enter a number: "))
gcd(a,b)
lcm(a,b)
```

```
Enter a number: 20
Enter a number: 24
GCD of 20 and 24 is 4
LCM of 20 and 24 is 120
```

Logic Gates

In [3]:

```
import boolean
algebra=boolean.BooleanAlgebra()
X,Y,F=algebra.symbols('X','Y','F')
X=bool(input("enter the input "))
Y=bool(input("enter the input "))
F=int(X & Y)
print("The output of AND gate is ",F)
```

```
enter the input 
enter the input 1
The output of AND gate is  0
```

In [4]:

```
import boolean
algebra=boolean.BooleanAlgebra()
X,Y,F=algebra.symbols('X','Y','F')
X=bool(input("enter the input "))
Y=bool(input("enter the input "))
F=int(X | Y)
print("The output of OR gate is ",F)
```

```
enter the input 
enter the input 1
The output of OR gate is  1
```

In [37]:

```
import boolean
algebra=boolean.BooleanAlgebra()
X=algebra.symbols('X')
F=algebra.symbols('F')
X=bool(input("enter the input "))
F=int(not X)
print("The output of NOT gate is ",F)
```

```
enter the input 1
The output of NOT gate is  0
```

In [22]:

```
import boolean
algebra=boolean.BooleanAlgebra()
X,Y,F=algebra.symbols('X','Y','F')
X=bool(input("enter the input "))
Y=bool(input("enter the input "))
F=int(X ^ Y)
print("The output of XOR gate is ",F)
```

```
enter the input 
enter the input 
The output of XOR gate is  0
```

Half Adder

In [28]:

```
import boolean
algebra=boolean.BooleanAlgebra()
X,Y,F,C=algebra.symbols('X','Y','F','C')
X=bool(input("enter the input "))
Y=bool(input("enter the input "))
F=int(X ^ Y)
C=int(X & Y)
print("The output of HALF ADDER gate is: ")
print("SUM: ",F)
print("CARRY: ",C)
```

```
enter the input 
enter the input 1
The output of HALF ADDER gate is: 
SUM:  1
CARRY:  0
```

Full Adder

In [34]:

```
import boolean
algebra=boolean.BooleanAlgebra()
X,Y,Z,F,C=algebra.symbols('X','Y','Z','F','C')
X=bool(input("enter the input "))
Y=bool(input("enter the input "))
Z=bool(input("enter the input "))
F=int(Z ^ (X ^ Y))
C=int((X & Y) | ((X ^ Y) & Z))
print("The output of FULL ADDER gate is: ")
print("SUM: ",F)
print("CARRY: ",C)
```

```
enter the input 
enter the input 1
enter the input 1
The output of FULL ADDER gate is: 
SUM:  0
CARRY:  1
```

Parallel Adder

In [42]:

```
import boolean
algebra=boolean.BooleanAlgebra()
X,Y,Z,V,G,F,C=algebra.symbols('X','Y','V','Z','G','F','C')
X=bool(input("enter the input "))
Y=bool(input("enter the input "))
Z=bool(input("enter the input "))
G=int(Z ^ (X ^ Y))
V=int((X & Y) | ((X ^ Y) & Z))
F=int(V ^ (X ^ Y))
C=int((X & Y) | ((X ^ Y) & V))
print("The output of PARALLEL ADDER gate is: ")
print("SUM 1: ",G)
print("CARRY out 1: ",V)
print("SUM out: ",F)
print("CARRY out: ",C)
```

```
enter the input
enter the input 1
enter the input 1
The output of PARALLEL ADDER gate is:
SUM 1:  0
CARRY out 1:  1
SUM out:  0
CARRY out:  1
```

In [1]:

```
#Program to find all the unique elements of a list.
def unique_list(l):
    x=[]
    for a in l:
        if a not in x:
            x.append(a)
    return x
print(unique_list ([10,20,30,5,8,4,20,10,4]))
```

```
[10, 20, 30, 5, 8, 4]
```

In [1]:

```
#Program to find all duplicates in the list.
def repeat(x):
    size=len(x)
    repeated=[]
    for i in range(size):
        k=1+i
        for j in range(k,size):
            if x[i]==x[j] and x[i] not in repeated:
                repeated.append(x[i])
    return repeated
list1=[50,10,10,10,20,20,30,40,50,50,60]
print(repeat(list1))
```

```
[50, 10, 20]
```

In [40]:

```
#Simple Calculator program by making use of functions
def addition(a,b):
    print("The sum of a & b is :",a+b)
def subtract(a,b):
    print("The difference of a & b is :",a-b)
def multiply(a,b):
    print("The product of a & b is :",a*b)
def divide(a,b):
    print("The division of a & b is :",a/b)
a=int(input("Enter the value of a :"))
b=int(input("Enter the value of b :"))
c=str(input("Enter the operation to perform :"))
if c=='+' :
    addition(a,b)
elif c=='-' :
    subtract(a,b)
elif c=='*' :
    multiply(a,b)
elif c=='/' :
    divide(a,b)
```

```
Enter the value of a :5
Enter the value of b :10
Enter the operation to perform :/
The division of a & b is : 0.5
```