fun and anomynous fun

function

sub program of main program is called fun python has 2 types of fun 1.fun definitiom 2.fun call Every fun call must have fun definition otherwise we get NameError. fun call use many times and fun def must define only once properties of fun 1.input : body call body call 2.process : body body body body 3.result/output : body call call body agr output call me aara toh body me retrn aata

In [15]:

```
#program 1
#input    :body
#process  :body
#output   :body

def sumop():
    a=int(input('enter a first number'))
    b=int(input('ente a second number'))
    c=a+b
    print('sum of {} and {}= {}'.format(a,b,c))
#main
sumop()
```

```
sum of 3 and 4= 7
```

In [19]:

```
#program 2
#input    :call
#process  :body
#output   :call

def sumop(a,b):  #a and b are formal paramater
    c=a+b      #c is local parameter
    return c
#main
a=int(input('enter a first value'))
b=int(input('enter a second value'))
res=a+b
print('sum of a nd b is {}'.format(res))
sumop(a,b)
```

```
sum of a nd b is 7
```

Out[19]:

```
7
```

In [29]:

```
#program 3
#input    :body
#process  :body
#output   :call

def sumop():
    a=int(input('enter a first value'))
    b=int(input('enter a second value'))
    c=a+b
    return a,b,c
#main
a,b,c=sumop()
print('sum of {} and {} ={} '.format(a,b,c))
```

```
sum of 4 and 56 =60
```

In [41]:

```
#program 3.2
#input    :body
#process  :body
#output   :call

def sumop():
    a=int(input('enter a first value'))
    b=int(input('enter a second value'))
    c=a+b
    return a,b,c
#main
a,b,c=sumop()
print('sum of {} and {} ={} '.format(a,b,c))
newsum=sumop()
print('sum of {} and {} is {}'.format(newsum[-2],newsum[-1],newsum[0]))
```

```
sum of 45 and 564 =609
sum of 645 and 1290 is 645
```

In [1]:

```
#program 4
#input    :call
#process  :body
#output   :body
```

```
def sum(a,b):
    c=a+b
    return c
    print('sum of a and b = {}'.format(c))
#main
a=int(input('enter a first number'))
b=int(input('enter a second number'))
sum(a,b)
```

Out[1]:

```
9
```

In [9]:

```
#program for simple interest
def simpleint():
    p=int(input('enter principale amount'))
    t=int(input('enter time '))
    r=int(input('enter rate of interest'))
    si=(p+t+r)/100
    tamt=p+si
    return p,t,r,si,tamt
#main
loanres=simpleint()
print('='*50)
print('SIMPLE INTEREST DETAIL')
print('='*50)
print('principale amount ={}'.format(loanres[0]))
print('time ={}'.format(loanres[1]))
print('rate ={}'.format(loanres[2]))
print('simple interest ={}'.format(loanres[3]))
print('total amount ={}'.format(loanres[4]))
```

```
==================================================
SIMPLE INTEREST DETAIL
==================================================
principale amount =60000
time =8
rate =8
simple interest =600.16
total amount =60600.16
```

In [21]:

```
#program for calculating factorial of number using fun
def calfact():
    n=int(input('enter a number for factorial'))
    if n<0:
        print('invalid')
    else:
        f=1
        for i in range(1,n+1):
            f=f*i
        else:
            print('fact {} = {}'.format(n,f))

#main
calfact()
```

```
fact 10 = 3628800
```

In [51]:

```
#program for list of word whose length is 3,4,5 and each word must contain  one vawel
def getnames():
    print('enter a names')
    namelist=[ names for  names in input().split()]
    return namelist
def namesw():
    namelist=getnames()
    for names  in namelist:
        print('{}'.format(names))
        vword=[name for name in namelist if (len(name) in [3,4,5] ) if ('a' in name.lower() or 'e' in name.lowe
                                                                       in name.lower() or 'u' in name.lower())]
        print('vwords')
        print(vword)
#main
namesw()
```

```
enter a names
shym
vwords
[]
```

In [49]:

```
#program for list of word whose length is 3,4,5 and each word must contain  one vawel
def getnames():
    print('enter a names')
    namelist=[ names for  names in input().split()]
    return namelist
```

```
def namesw():
    namelist=getnames()
    for names  in namelist:
        print('{}'.format(names))
        vword=[name for name in namelist if (len(name) in [3,4,5] ) if ('a' in name.lower() or 'e' in name.lowe
                                                                       in name.lower() or 'u' in name.lower())]
        print('vwords')
        print(vword)
#main
namesw()
```

```
enter a names
ali
vwords
['ali']
```

anomynous fun

-anomynous fun also called lambda fun ,because itzz used lamnda kwd. -used to perform instant operations. -contain single executable statement.

In [17]:

```
#program for adding values using normanl fun and anomynous fun
def addop(a,b):
    addop=lambda a,b:a+b
    c=a+b
    return c
#main
res=addop(10,20)
res1=addop(100,200)
print(res)
print(res1)
```

```
30
300
```

In [21]:

```
def sum(a,b):
    sum=lambda a,b:a+b
    c=a+b
    return c
x=int(input('enterr first number'))
y=int(input('enterr second number'))
res=sum(x,y)
result=sum(x,y)
print(res)
print(result)
```

```
70
70
```

In [ ]:

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js